

PB86-137460

INITIAL MIXING CHARACTERISTICS OF MUNICIPAL OCEAN
DISCHARGES: VOLUME II, COMPUTER PROGRAMS

U.S. Environmental Protection Agency
Newport, OR

Nov 85



U.S. Department of Commerce
National Technical Information Service
NTIS.

United States
Environmental Protection
Agency

Environmental Research
Laboratory
Narragansett RI 02882

EPA/600/3-85/073b
November 1985

Research and Development

PB86-137460



Initial Mixing Characteristics of Municipal Ocean Discharges:

Volume II. Computer Programs

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U.S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA. 22161

REPORT DOCUMENTATION PAGE		1. REPORT NO. EPA/SW/MT-86/012b	2.	3. Recipient's Accession No. PB86-137460
4. Title and Subtitle Initial Mixing Characteristics of Municipal Ocean Discharge Volume II: Computer Programs			5. Report Date	
7. Author(s) M.D. Schuldt, L.R. Davis, and W.E. Frick W. P. Muellenhoff, A. M. Soldate, Jr. D.J. Baumgartner			6.	
9. Performing Organization Name and Address Pacific Division, Environmental Research Lab. Narragansett Office of Records & Development U. S. Environmental Protection Agency Newport, OR 97365			8. Performing Organization Rept. No.	
12. Sponsoring Organization Name and Address			10. Project/Task/Work Unit No.	
15. Supplementary Notes			11. Contract(C) or Grant(G) No. (C) (G)	
For magnetic tape see PB86-137494 For Diskette see PB86-137486			13. Type of Report & Period Covered	
16. Abstract (Limit: 200 words)			14.	
This report (Volume II) contains a description of the Universal Data File and complete program listings of five mathematical models that provide flux-average dilution and rise-height of a wastewater plume discharged into waters of greater density. The companion report (Volume I) contains analytical solutions and detailed descriptions of the models. Guidance is provided for the range of values within which the analytical solutions provide acceptable estimates. Use of the models is recommended for conditions outside these ranges and for detailed analysis.				
17. Document Analysis a. Descriptors				
b. Identifiers/Open-Ended Terms				
c. COSATI Field/Group				
18. Availability Statement		19. Security Class (This Report)	21. No. of Pages 108	
		20. Security Class (This Page)	22. Price	

EPA-600/3-85-073b
November 1985

INITIAL MIXING CHARACTERISTICS OF MUNICIPAL OCEAN DISCHARGES
VOLUME II - COMPUTER PROGRAMS

by

W.P. Muellenhoff, A.M. Soldate, Jr., D.J. Baumgartner
M.D. Schuldt, L.R. Davis, and W.E. Frick

PACIFIC DIVISION
ENVIRONMENTAL RESEARCH LABORATORY, NARRAGANSETT
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
NEWPORT, OREGON 97365

i-o

The information in this document has been funded by the United States Environmental Protection Agency Office of Research and Development, and the Office of Marine and Estuarine Protection through contract numbers 68-01-5906 and 68-01-6922 to Tetra Tech, Inc. Agency Project Officers for these contracts are Dr. John Pai and Mr. Barry Burgan, respectively. The report has been subject to the Agency's peer and administrative review, and it has been approved as an EPA document.

FORWARD

A portion of this document is based on an earlier version by A.M. Teeter and D.J. Baumgartner (1979), which it supersedes. The technical reviews and resultant suggestions of A.R. Agg, W.A. Faisst, Irwin Haydock and P.J. Roberts resulted in many improvements and are gratefully acknowledged. We are also thankful to S.J. Wright, V.H. Chu, and other scientists who have indirectly contributed to the report in the form of fruitful dialogue during its development. Their continued inputs are encouraged, and will ensure timely publication of addenda and further improvements in future editions.

W.P. Muellenhoff is the Director, Corvallis Office, Tetra Tech, Inc., and A.M. Soldate, Jr. is a Senior Scientist in Environmental Systems Engineering at Tetra Tech, Inc., Bellevue, WA. D.J. Baumgartner, M.D. Schuldt, and W.E. Frick are with the U.S. Environmental Protection Agency, Pacific Division (Newport, OR). L.R. Davis is Professor, Mechanical Engineering Department, Oregon State University.

Users of this document or the models described herein are encouraged to report any errors to enable appropriate corrections to be made. Direct all correspondence to D.J. Baumgartner, U.S. Environmental Protection Agency, Hatfield Marine Science Center, Newport, Oregon 93765. Holders of the document should notify the above to receive errata or future revisions to the document.

ABSTRACT

This report (Volume II) contains a description of the Universal Data File and complete program listings of five mathematical models that provide flux-average dilution and rise-height of a wastewater plume discharged into waters of greater density. The companion report (Volume I) contains analytical solutions and detailed descriptions of the models. Guidance is provided for the range of values within which the analytical solutions provide acceptable estimates. Use of the models is recommended for conditions outside these ranges and for detailed analysis.

Volumes I and II are available in hardcopy from the National Technical Information Service (5285 Port Royal Road, Springfield, VA, 22161; 703-487-4650). Volume II is also available from NTIS on a 9-track tape or diskette (703-487-4763). The IBM-PC compatible diskette has the programs slightly altered to compile using Microsoft FORTRAN (Version 3.1 or higher) or IBM Personal Computer Professional FORTRAN (8087 or 80287 chip required).

CONTENTS

	<u>Page</u>
Forward	iii
Abstract	iv
SECTIONS	
1. Universal Data File	1
2. UPLUME Listing	4
3. UOUTPLM Listing	22
4. UDKHDEN Listing	38
5. UMERGE Listing	72
6. ULINE Listing	89

SECTION 1
Universal Data File

UNIVERSAL DATA FILE (UDF) "CARD" DECK

THE DATA ENTERED ON CARDS 2 THROUGH 7 MAY BE EITHER IN THE
FORMAT REQUIRED BY EACH CARD OR EACH VALUE ON THE CARD MAY
BE SEPARATED BY A COMMA (SHORT FIELD TERMINATION).
AN EXPLICIT DECIMAL POINT OVERRIDES THE FIELD DESCRIPTOR.

CARD 1 FORMAT(10A8)
IDENTIFICATION OF A DATA SET WITHIN THE UDF.

CARD 2 FORMAT(8I2)
INTER =1 INTERACTIVE CONTROL OF CARDS 3 AND 4 PARAMETERS.
=0 "SINGLE" RUN USING PARAMETERS IN DATA SET ONLY.
IDFP =1 PRINT "CARD IMAGE" OF DATA SET.
=0 DO NOT PRINT CARD IMAGE OF DATA SET.
ICUTOP =1 USE OPTIONAL CARD 5 TO CHANGE CONTROL PARAMETERS FROM
THE DEFAULT VALUES.
=0 DO NOT READ A CARD 5 (THUS CARD 5 MUST BE OMITTED).

IPI	INPUT PRINTOUT CONTROL FOR	UPLUME
IOI	"	UOUTPLM
IDI	"	UDKHEN (SEE NOTE 1)
IMI	"	UMERGE
ILI	"	ULINE

IPO=IPI	OUTPUT PRINTOUT CONTROL FOR	UPLUME
IOO=IOI	"	UOUTPLM
IDO=IDI	"	UDKHEN (SEE NOTE 1)
IMO=IMI	"	UMERGE
ILO=ILI	"	ULINE

FOR EACH OF THE PARAMETERS IPI TO ILI
=0 USE NEW (8.5 X 11) FORMAT.
=1 USE ORIGINAL FORMAT.
=2 USE CONDENSED FORMAT (USEFUL IN INTERACTIVE MODE).

NOTE! 1) IDI AND IDO ALLOWED FOR BUT PRESENTLY NOT USED
IN UDKHEN, ENTER THE SAME VALUE AS THE OTHERS.

CARD 3 FORMAT(F10.0,I10,3F10.0)
QT TOTAL EFFLUENT FLOW (CUBIC METERS PER SEC).
NP NUMBER OF PORTS (SEE NOTE 2).
PDIA PORT DIAMETER (M), EFFECTIVE DIAMETER IF KNOWN.
VANG VERTICAL ANGLE (DEG) OF PORT RELATIVE TO THE
HORIZONTAL (90 DEGREES FOR A VERTICAL PORT).
ULINE ASSUMES VANG=90 DEG.
PDEP PORT DEPTH (M) MUST BE GREATER THAN 0.0 AND
LESS THAN OR EQUAL TO THE DEEPEST DEPTH OF THE
AMBIENT DENSITY PROFILE.

NOTE! 2) ULINE REQUIRES TWO OR MORE PORTS, FOR THE
OTHERS, IF NP=1 SPACE=1000.0 (DEFAULT) MAKING
THE MERGING FLAGS INACTIVE.

CARD 4 FORMAT(3F10.0)
UM HORIZONTAL CURRENT SPEED (M/S) (USED IN UOUTPLM ONLY).
HANG ANGLE (DEG) OF CURRENT DIRECTION WITH RESPECT TO DIFFUSER
AXIS (90 DEGREES CORRESPONDS TO A CURRENT DIRECTION
PERPENDICULAR TO THE DIFFUSER AXIS AND IF VANG=0, BOTH
THE CURRENT AND THE DISCHARGE ARE IN THE SAME DIRECTION)
(SEE NOTE 3).
SPACE DISTANCE (M) BETWEEN ADJACENT PORTS (SEE NOTE 2).

NOTE! 3) HANG NOT USED IN UPLUME. UOUTPLM AND UMERGE
ASSUME 90 DEG. UDKHEN RANGE 45 - 135 DEG FOR
MORE THAN ONE PORT AND 0 - 180 DEG FOR A SINGLE
PORT (NOTE, SINGLE PORT ONLY: FOR VALUES GREATER

THAN 90 DEG BUT LESS THAN OR EQUAL TO 180 DEG, THE
PROGRAM SETS HANG EQUAL TO THE SUPPLEMENTARY ANGLE).
ULINE RANGE 0 - 180 DEG.

CARD 5 OPTIONAL (INCLUDE THIS CARD ONLY IF ICUTOP =1)
FORMAT(F5.0,2I5,3I2,6F5.0,2I5)

USED IN UMERGE

A	ASPIRATION COEFFICIENT	0.1	BY DEFAULT
ITER	MAXIMUM NUMBER OF ITERATIONS	5000	BY DEFAULT
IFRQ	ITERATION PRINTOUT FREQUENCY	150	BY DEFAULT
NAA	PRINT ARRAY AA IF =1, DO NOT IF =0	0	BY DEFAULT
NAB	PRINT ARRAY AB IF =1, DO NOT IF =0	0	BY DEFAULT
NAC	PRINT ARRAY AC IF =1, DO NOT IF =0	0	BY DEFAULT

(SEE LISTING OF PROGRAM UMERGE FOR CONTENTS OF ARRAYS
AA, AB, AC WHICH ARE MAINLY DEBUGGING AIDS.)

USED IN UPLUME

PS	PRINTOUT "INTERVAL"	3.	BY DEFAULT
----	---------------------	----	------------

USED IN ULINE

RK	RATIO OF SA/SM IN ROBERTS' EXPERIMENTS	1.41	BY DEFAULT
DM	INTEGRATION STEP SIZE (M)	0.1	BY DEFAULT

USED IN UOUTPLM

H	INITIAL THICKNESS OF PLUME ELEMENT	.5*PDIA	BY DEFAULT
E	IMPINGEMENT ENTRAINMENT COEFFICIENT	1.0	BY DEFAULT
A	ASPIRATION ENTRAINMENT COEFFICIENT	0.1	BY DEFAULT
ITERB	NUMBER OF INTEGRATION STEPS ALLOWED	5000	BY DEFAULT
IR	PRINTOUT INTERVAL	50	BY DEFAULT

NOTE! WHEN CARD 5 IS USED, ALL OF THE PARAMETERS NEED NOT BE
GIVEN A NEW VALUE, ONLY THE ONES TO BE CHANGED. ENTER ZERO
FOR THE OTHERS AND THERE DEFAULT VALUES WILL BE USED.

ITER, IFRQ, ITERB AND IR NOT TO EXCEED FOUR DIGITS.

NO OPTIONS AVAILABLE FOR UDKHDEN.

CARD 6 FORMAT(I10,2F10.0)

NPTS	NUMBER OF DEPTHS WHERE AMBIENT TEMPERATURE, SALINITY, AND HORIZONTAL CURRENT SPEED ARE KNOWN (NPTS MUST BE AT LEAST EQUAL TO 2 AND NOT MORE THAN 30).
S	EFFLUENT SALINITY (PPT) IF T NOT EQUAL TO ZERO
T	EFFLUENT DENSITY (G/CM3) IF T=0 EFFLUENT TEMPERATURE (DEGREES CELSIUS). IF T=0 PROGRAMS ASSUME S IS EFFLUENT DENSITY IN G/CM3, SEE NOTE 4.

CARD 7 FORMAT(4F10.0)

DP()	DEPTH IN METERS, MUST HAVE DATA FOR DP()=0.0
SA()	AMBIENT SALINITY (PPT) IF TA() NOT EQUAL TO ZERO
TA()	AMBIENT DENSITY (G/CM3) IF TA()=0 AMBIENT TEMPERATURE (DEGREES CELSIUS). IF TA()=0 PROGRAMS ASSUME SA() IS AMBIENT DENSITY IN G/CM3, SEE NOTE 4.
UA()	HORIZONTAL AMBIENT CURRENT SPEED (M/S) (USED IN UMERGE, UDKHDEN, AND ULINE).

NOTE! 4) THERE MUST BE NPTS IMAGES OF CARD 7. ALSO, EITHER
ALL TA(I) MUST BE ZERO OR ALL NOT ZERO, OR ERRORS
IN THE INTERPRETATION OF SA() AND TA() WILL OCCUR.
IF, FOR SOME I, SA(I) IS DESIRED TO REPRESENT
AMBIENT SALINITY AND TA(I) SHOULD BE EXACTLY 0, SET
TA(I) EQUAL TO A SMALL NUMBER INSTEAD (0.000001 FOR
INSTANCE). THIS APPLIES TO S AND T AS WELL.

SECTION 2

UPLUME Listing

PROGRAM UPLUME

```

1  PROGRAM UPLUME
2  C
3  C  UP-TO-DATE AS OF AUGUST 1985 (NO CHANGES SINCE 7-26-85).
4  C  JULY 26, 1985 MODIFIED OUTPUT WHEN PLUMES MERGE AFTER TRAPPING.
5  C  JULY 21, 1985 REVISED TO TERMINATE SOME COMPUTATIONS AT 0.9*DEPTH
6  C  TO ACCOMMODATE BLOCKING NEAR THE WATER SURFACE.
7  C  JUNE 17, 1985 REVISED INTERACTIVE DATA INPUT SO USER MAY CORRECT
8  C  ERRORS BEFORE RERUNNING THE PROGRAM.
9  C  MAY 15, 1985 DELETED PRINTING OF TIME AND PLUME DIAMETER WHEN
10 C  PRINTING OF DILUTION CEASES. REVISED OUTPUT WHEN MERGING OCCURS
11 C  BEFORE TRAPPING.
12 C  MARCH 27, 1985 CHANGE ANGLE OF DISCHARGE LIMITS FROM 0 TO 90 DEG
13 C  TO -5 TO 90 DEG.
14 C  MARCH 19, 1985 RELOCATED THE INTERACTIVE RECYCLE STATEMENT
15 C  (3 WRITE) TO 3 CONTINUE, THE FIRST STATEMENT UNDER INTERACTIVE
16 C  CONTROL. ADDED STATEMENTS IN SUBROUTINE LIMITS TO DETECT
17 C  ERRONEOUS RESPONSES TO PROGRAM PROMPTS I.E. .NE. TO YES OR NO.
18 C
19 C  PLUME MODIFIED TO USE UNIVERSAL DATA FILE WITH OPTIONS
20 C
21 C  SUBROUTINES RPLUME, MPLUME, LIMITS, USORT, SIGMAT, SDERIV
22 C
23 C  COMMON G,FK,FH,COSTH,SINTH,COSTHE,DS,C1,C2,E13,FLAG,GRAV
24 C  *,ICEASE
25 C  COMMON/AVAR/DP(30),SA(30),TA(30),RHO(30),DENPP(30)
26 C  COMMON/VAR/INTER,IDFP,ICUTOP,IPSF,IPI,IPO,ANGLE,Q,DIA,
27 C  *DEPTH,RHOJ,FN,PS,SPACE,SI,TI,NPTS,RFDD,UO,NQ
28 C  COMMON/COUNT/NWO,NW
29 C  COMMON/INOUT/IN,IO,IT
30 C  DIMENSION ZD(30),DG(30)
31 C  REAL*8 TITLE(10),TITLI(10)
32 C  INTEGER CHGDEN,ANS,TRAPPD,FLAG,BLOC
33 C  BYTE IFILE(20),OFILE(20)
34 C  COMMON/TVAR/TITLE,TITLI,IFILE
35 C  COMMON CUBRT,TWOTH
36 C  DATA IFILE(20)/O/,OFILE(20)/O/
37 C
38 C:::::INITIALIZE CONSTANTS AND A VARIABLE.
39 C
40 C  IN THIS PROGRAM (PDP SYSTEM - FORTRAN IV-PLUS),
41 C  LOGICAL UNIT IN IS THE INPUT FILE.
42 C  " " IO IS THE OUTPUT FILE.
43 C  " " IT IS THE USER TERMINAL.
44 C
45 C  IN=1
46 C  IO=2
47 C  IT=5
48 C
49 C  NWO=0
50 C  NW=0
51 C  GRAV=9.81
52 C  CUBRT=1./3.
53 C  TWOTH=2./3.
54 C  NOCASE=0
55 C
56 C:::::OPEN FILES
57 C
58 C  WRITE(IT,554)
59 C  554 FORMAT(1X,' *****PROGRAM UPLUME, AUGUST 1985*****')
60 C  WRITE(IT,555)
61 C  555 FORMAT(1X,' ENTER UDF NAME: ',S)
62 C  READ(IT,556) (IFILE(IK),IK=1,19)
63 C  556 FORMAT(19A1)
64 C  WRITE(IT,557)
65 C  557 FORMAT(1X,' ENTER OUTPUT FILE NAME: ',S)

```

PROGRAM UPLUME

```

66      READ(IT,556)(OFILE(IK),IK=1,19)
67      OPEN(UNIT=IN,NAME=IFILE,TYPE='OLD')
68      OPEN(UNIT=IO,NAME=OFILE,TYPE='NEW')
69      C
70      C:::::85 IS RECYCLE STATEMENT. SET/RESET COUNTERS, INITIALIZE
71      C          LIMIT FLAG.
72      C
73      85 CONTINUE
74      NWO=NWO+NW
75      NCW=0
76      LIMIT=0
77      C
78      C:::::READ/SORT/CHECK LIMITS OF DATA SET.
79      C
80      CALL RPLUME(1,LIMIT)
81      CALL USORT(DP,SA,TA,RHO,NPTS)
82      CALL LIMITS(1,DEPTH,ANGLE,TI,SI,NPTS,INTER,TITLE,TITLI,
83      *IEXIT,LIMIT)
84      IF(IEXIT .EQ. 0)GO TO 21
85      CALL RPLUME(2,LIMIT)
86      GO TO 85
87      C
88      C:::::IF DATA CHECKS OK, START COMPUTATIONS.
89      C
90      21 CONTINUE
91      C
92      C:::::IS AMBIENT ENTERED AS DENSITY OR TEMP. AND SALINITY?
93      C
94      IFLAG=1
95      IF(TA(NPTS) .EQ. 0.)IFLAG=0
96      IF(IFLAG .EQ. 0)GO TO 1
97      C
98      C:::::IT'S TEMP. AND SALINITY, COMPUTE DENSITY AND INITIALIZE
99      C          DENSITY AND SIGMA T ARRAYS.
100     C
101     DO 1000 I=1,NPTS
102     DENPP(I)=SIGMAT(SA(I),TA(I))
103     RHO(I)=1.+0.001*DENPP(I)
104     1000 CONTINUE
105     GO TO 2
106     C
107     C:::::IT'S DENSITY, INITIALIZE DENSITY AND SIGMA T ARRAYS.
108     C
109     1 DO 1001 I=1,NPTS
110     RHO(I)=SA(I)
111     DENPP(I)=(SA(I)-1.)*1000.
112     1001 CONTINUE
113     2 CONTINUE
114     KPAGE=0
115     C
116     C:::::IF ONLY ONE DISCHARGE PORT, PORT SPACING IS SET
117     C          EQUAL TO 1000. TO PREVENT PLUMES MERGING FLAG.
118     C
119     IF(FN .EQ. 1.)SPACE=1000.
120     IF(INTER .EQ. 0)GO TO 20
121     C
122     C:::::INTERACTIVE CONTROL.
123     C          3 IS THE RECYCLE STATEMENT IN THE INTERACTIVE (INTER=1) MODE.
124     C
125     3 CONTINUE
126     IF(NCW .EQ. 0)GO TO 11
127     WRITE(IT,10)(TITLE(IK),IK=1,10)
128     10 FORMAT(/1X,' DO YOU WISH TO CHANGE ANY INPUT VARIABLES AND RERUN?'
129     */1X,' CASE I.D. ',10A8/1X,' YES TO RERUN OR NO FOR NEXT CASE: ',3)
130     READ(IT,9)ANS

```

PROGRAM UPLUME

```

131      9 FORMAT(A1)
132      IF(ANS .EQ. 'Y')GO TO 11
133      IF(ANS .EQ. 'N')GO TO 85
134      WRITE(IT,12)
135      12 FORMAT(/1X,' ***** YOU MADE A MISTAKE, TRY AGAIN *****')
136      GO TO 3
137      11 CONTINUE
138      IF(NCW .EQ. 0)WRITE(IT,7)(TITLE(IK),IK=1,10)
139      7 FORMAT(/1X,' CASE IDENTIFICATION: '/
140      *2X,10A8)
141      WRITE(IT,13)
142      13 FORMAT(1X,' WHAT IS THE TITLE OF THIS INTERACTIVE RUN? '/
143      *2X,S)
144      READ(IT,6)(TITLI(IK),IK=1,10)
145      6 FORMAT(10A8)
146      IF(NCW .EQ. 0)GO TO 22
147      14 CONTINUE
148      NP=FN
149      WRITE(IT,110)Q,NP,DIA,SPACE,ANGLE,DEPTH
150      110 FORMAT(/1X,' YOUR PRESENT INPUT VARIABLES ARE: '//
151      *1X,' 1. EFFLUENT FLOW (CMS)          Q =',F10.4/
152      *1X,' 2. NUMBER OF PORTS              NP =',I5/
153      *1X,' 3. PORT DIAMETER (M)            DIA =',F10.4/
154      *1X,' 4. PORT SPACING (M)             SPACE =',F9.3/
155      *1X,' 5. VERTICAL ANGLE W/HORZ (DEG) ANGLE =',F9.3/
156      *1X,' 6. PORT DEPTH (M)              DEPTH =',F8.2)
157      IF(TI .EQ. 0.)WRITE(IT,111)SI
158      111 FORMAT(
159      *1X,' 7. EFFLUENT DENSITY (G/CM3)      SI =',F11.5)
160      IF(TI .NE. 0.)WRITE(IT,112)SI,TI
161      112 FORMAT(
162      *1X,' 7. EFFLUENT SAL (0/00)           SI =',F8.2/
163      *1X,' 8. EFFLUENT TEMP (DEG C)         TI =',F8.2)
164      WRITE(IT,113)
165      113 FORMAT(/1X,' ENTER THE NUMBER OF THE VARIABLE
166      * YOU WISH TO CHANGE. ',S)
167      READ(IT,115)NUMBER
168      115 FORMAT(I1)
169      IF(TI .EQ. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 7)GO TO 15
170      IF(TI .NE. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 8)GO TO 15
171      WRITE(IT,12)
172      GO TO 14
173      15 CONTINUE
174      WRITE(IT,16)
175      16 FORMAT(1X,' WHAT IS THE VALUE OF THIS VARIABLE? ',S)
176      READ(IT,18)VALUE
177      18 FORMAT(F10.0)
178      IF(NUMBER .EQ. 1)Q=VALUE
179      IF(NUMBER .EQ. 2)FN=VALUE
180      IF(NUMBER .EQ. 3)DIA=VALUE
181      IF(NUMBER .EQ. 4)SPACE=VALUE
182      IF(FN .EQ. 1.)SPACE=1000.
183      IF(NUMBER .EQ. 5)ANGLE=VALUE
184      IF(NUMBER .EQ. 6)DEPTH=VALUE
185      IF(NUMBER .EQ. 7)SI=VALUE
186      IF(NUMBER .EQ. 8)TI=VALUE
187      19 CONTINUE
188      WRITE(IT,17)
189      17 FORMAT(1X,' DO YOU WISH TO CHANGE ANOTHER VARIABLE?
190      * YES OR NO: ',S)
191      READ(IT,9)ANS
192      IF(ANS .EQ. 'N')GO TO 8
193      IF(ANS .EQ. 'Y')GO TO 14
194      WRITE(IT,12)
195      GO TO 19

```

PROGRAM UPLUME

```

196      8 CONTINUE
197      CALL LIMITS(2,DEPTH,ANGLE,TI,SI,NPTS,
198      *INTER,TITLE,TITLI,EXIT,LIMIT)
199      IF(IEEXIT.EQ. 0)GO TO 22
200      CALL RPLUME(2,LIMIT)
201      GO TO 85
202      22 NCM=1
203      C
204      C:::::SET INITIAL CONDITIONS.
205      C
206      20 CONTINUE
207      NQ=FM
208      NOCASE=NOCASE+1
209      RFD=0.
210      FD=0.
211      C
212      C:::::IS THE EFFLUENT ENTERED AS DENSITY OR TEMP. AND SALINITY?
213      C
214      IF(TI.EQ. 0.)RHOJ=SI
215      IF(TI.NE. 0.)RHOJ=1.+0.001*SIGMAT(SI,TI)
216      C
217      C:::::FIND THE PROFILE DEPTH THAT IS EQUAL TO OR GREATER
218      C      THAN THE DEPTH OF DISCHARGE.
219      C
220      DO 1200 I=1,NPTS
221      IF(DP(I).GE. DEPTH)GO TO 32
222      1200 CONTINUE
223      32 NP=I
224      NM=I-1
225      C
226      C:::::COMPUTE THE DENSITY AT THE DEPTH OF DISCHARGE (RHOJ) AND THE
227      C      DIFFERENCE (DISP) BETWEEN THE EFFLUENT (RHOJ) AND THE
228      C      AMBIENT AT THE DEPTH OF DISCHARGE.
229      C
230      RHOJ=(DEPTH-DP(NM))*(RHO(NP)-RHO(NM))/(DP(NP)-DP(NM))+RHO(NM)
231      DISP=RHOJ-RHOJ
232      C
233      C:::::INITIALIZE ZD & DG ARRAYS.
234      C
235      DO 1201 I=1,NM
236      J=NP-I
237      ZD(I)=(DEPTH-DP(J))/DIA
238      DG(I)=(RHO(J+1)-RHO(J))*DIA/(DISP*(DP(J+1)-DP(J)))
239      1201 CONTINUE
240      C
241      C:::::COMPUTE THE DISCHARGE VELOCITY (UO) AND THE SQUARE OF THE
242      C      FROUDE NUMBER (RFD). RFD IS THE SQUARE ROOT OF RFD.
243      C
244      UO=Q/(FM*.7853982*DIA*DIA)
245      RFD=UO*UO*RHOJ/(-DISP*DIA*GRAV)
246      FD=ABS(RFD)
247      RFD=SQRT(FD)
248      C
249      C:::::COMPUTE LENGTH FOR FLOW ESTABLISHMENT (SP),
250      C      A FUNCTION OF THE FROUDE NUMBER.
251      C
252      IF(FD.LT. 4. .OR. FD.GT. 10.24)GO TO 34
253      S=.113*FD+4.
254      GO TO 35
255      34 IF(FD.LT. 4.)S=2.8*FD**CUBRT
256      IF(FD.GT. 10.24)S=5.6*FD/SQRT(FD*FD+18.)
257      SP=S*DIA
258      C
259      C:::::COMPUTE STARTING ANGLE IF DISCHARGE ANGLE IS NOT 90 DEG.
260      C

```

PROGRAM UPLUNE

```

261      35 TEMP=ATAN(1.416667*S/FD)
262      THETAO=(ANGLE/90.)*(1.5708-TEMP)+TEMP
263      COSTHE=COS(THETAO)
264      SIN THE=SIN(THETAO)
265      C
266      C:::::COMPUTE INTEGRATION STEP LENGTH (DSI).
267      C
268          DSI=DEPTH/(177.*DIA)
269          IF(DG(1) .EQ. 0.)GO TO 36
270          DGTEMP=.01/DG(1)
271          IF(DGTEMP .LE. 0.)DGTEMP=-DGTEMP
272          DSI=.12*1.6**(ALOG10(FD/10.))*2.**(ALOG10(DGTEMP))
273      C
274      C:::::INITIALIZE PRINTOUT CONTROL (NPO) AND ADDITIONAL
275      C          VARIABLES AND CONTROL FLAGS.
276      C
277          36 NPO=IFIX(PS/(DSI*DIA)+.5)
278          N=0
279          Z=S*SIN THE
280          X=S*COSTHE
281          E=(4./S)**3
282          R=.25
283          C1=E**TWO THD
284          C2=.75/RFD
285          IPTS=1
286          G=DG(IPTS)
287          ZLIP=ZD(IPTS)
288          CHGDEN=0
289          FLAG=0
290          TRAPP=0
291          XP=.*DIA
292          ZP=DEPTH-Z*DIA
293          DSI P=DSI*DIA
294          ZRMAX=.9*DEPTH/DIA
295      C
296      C:::::T IS INITIAL TIME STEP.
297      C
298          T=2.*SP/(UD*(1.+36.77/(S*S)))
299      C
300      C:::::WRITE INITIAL CONDITIONS
301      C
302          IF(IPI .EQ. 1)GO TO 45
303          IF(KPAGE .EQ. 1)GO TO 41
304          CALL RPLUME(2,LIMIT)
305          WRITE(IO,130)
306          130 FORMAT(1H1,' UPLUNE VERSION 1.0 AUGUST 1985 (BASED ON OS3
307          * VERSION 2.3 9/12/77)',/1X,4(18(' '),2X))
308          WRITE(IO,561)(IFILE(IK),IK=1,19)
309          561 FORMAT(1X,' UNIVERSAL DATA FILE: ',19A1)
310          WRITE(IO,131)(TITLE(IK),IK=1,10)
311          131 FORMAT(2X,'CASE I.D. ',10A8)
312          IF(INTER .EQ. 1)WRITE(IO,134)(TITLI(IK),IK=1,10)
313          134 FORMAT(2X,'RUN TITLE: ',10A8)
314          CALL NPLUME(1,IFLAG)
315          GO TO 43
316          41 IF(IPI .EQ. 2)GO TO 42
317          CALL RPLUME(2,LIMIT)
318          WRITE(IO,130)
319          WRITE(IO,561)(IFILE(IK),IK=1,19)
320          WRITE(IO,131)(TITLE(IK),IK=1,10)
321          WRITE(IO,134)(TITLI(IK),IK=1,10)
322          CALL NPLUME(1,IFLAG)
323          GO TO 43
324          42 CALL NPLUME(3,IFLAG)
325          43 IF(IPO .EQ. 0)WRITE(IO,133)

```

PROGRAM UPLUME

```

326 133 FORMAT(/72X,'FLUX-AVE',/7X,'T',9X,'S',9X,'X',9X,'Z',
327 *8X,'DIA',8X,'H',7X,'THETA',2X,'DILUTION',/5X,'(SEC)',
328 *6X,'(M)',7X,'(M)',7X,'(M)',7X,'(M)',7X,'(M)',6X,'(DEG)')
329 GO TO 46
330 45 CALL RPLUME(2,LIMIT)
331 WRITE(10,200)
332 200 FORMAT(1H1,1X,'OS3 PLUME VERSION 2.3 9/12/77
333 * (MODIFIED FOR UNIVERSAL DATA FILE, AUGUST 1985.)')
334 WRITE(10,201)
335 201 FORMAT(1X,'
336 * *****A BUOYANT PLUME IN A
337 * DENSITY STRATIFIED MEDIA*****')
338 WRITE(10,561)(IFILE(IK),IK=1,19)
339 WRITE(10,205)(TITLE(IK),IK=1,10)
340 205 FORMAT(1X,' CASE I.D. ',10A8)
341 IF(INTER.EQ.1)WRITE(10,134)(TITLI(IK),IK=1,10)
342 WRITE(10,206)NOCASE
343 206 FORMAT(/1X,' CASE NO.',I3,' WITHIN THE UDF, UNITS: MKS,
344 * INITIAL CONDITIONS....')
345 WRITE(10,208)ANGLE,RFD0,SP,DSIP,PS
346 208 FORMAT(
347 *40H PORT ANGLE . . . . .,F7.1/
348 *40H FROUDE NUMBER. . . . .,F7.1/
349 *40H LENGTH FOR FLOW ESTABLISHMENT. . . ,F8.2/
350 *40H INTEGRATION STEP LENGTH. . . . .,F9.3/
351 *40H PRINTOUT INTERVAL. . . . .,F8.2)
352 WRITE(10,308)XP,ZP,RHOJ,DEPTH
353 308 FORMAT(
354 *40H XO . . . . .,F8.2/
355 *40H ZO . . . . .,F8.2/
356 *40H DISCHARGE DENSITY. . . . .,F11.5/
357 *40H PORT DEPTH . . . . .,F8.2)
358 WRITE(10,209)Q,NQ,UO,DIA
359 209 FORMAT(
360 *40H FLOWRATE . . . . .,F10.4/
361 *40H NUMBER OF PORTS. . . . .,I5/
362 *40H DISCHARGE VELOCITY . . . . .,F8.2/
363 *40H PORT DIAMETER. . . . .,F10.4)
364 IF(NQ.GT.1)WRITE(10,202)SPACE
365 202 FORMAT(
366 *40H PORT SPACING . . . . .,F9.3)
367 IF(NQ.EQ.1)WRITE(10,203)
368 203 FORMAT(4X,'PORT SPACING . . . . . 1000.0 (DEFAULT)')
369 WRITE(10,210)((DP(I),RHO(I)),I=1,NPTS)
370 210 FORMAT(/1X,' DENSITY STRATIFICATION: DEPTH RHO'//
371 *(26X,F6.2,F11.5))
372 C
373 C:::::MAIN COMPUTATIONAL SCHEME
374 C
375 46 DS=DSI
376 ICEASE=0
377 IMR=0
378 BLOC=0
379 NBLOC=0
380 NTRAP=0
381 NMERGE=0
382 BNEW=.308*SP
383 ASPAC=SPACE
384 IF(IPO.EQ.1)WRITE(10,211)
385 211 FORMAT(/1X,6X,'T',9X,'S',9X,'X',9X,'Z',9X,'D',8X,'ELEV',
386 *5X,'THETA',3X,'DILN(CL)',2X,'DILN(AVE)')
387 GO TO 55
388 51 DS=DSI
389 52 DELX=COSTH*DS
390 DELZ=SINTH*DS

```

PROGRAM UPLUME

```

391     DELE=FK
392     DELT=FM
393     SPDS2=S+DS*.5
394 C
395 C:::::4TH ORDER RUNGE-KUTTA INTEGRATION.
396 C
397     DO 2000 I=1,2
398     CALL SDERIV(SPDS2,E+.5*FK,R+.5*FM)
399     DELX=DELX+2.*COSTH*DS
400     DELZ=DELZ+2.*SINTH*DS
401     DELE=DELE+2.*FK
402     DELT=DELT+2.*FM
403     2000 CONTINUE
404 C
405     CALL SDERIV(S+DS,E+FK,R+FM)
406     ZLAST=Z
407     ZINCR=(DELZ+SINTH*DS)/6.
408     Z=Z+ZINCR
409     IF (CHGDN .EQ. 1)GO TO 71
410     IF (Z .GT. ZLIM)GO TO 70
411     53 CONTINUE
412     OLDX=X
413     X=X+(DELX+COSTH*DS)/6.
414     EOLD=E
415     E=E+(DELE+FK)/6.
416     R=R+(DELT+FM)/6.
417     IF(E .LE. 0.)E=0.
418     DT=.218*DIA*DS/(UD*(EOLD**CUBRT/S+E**CUBRT/(S+DS)))
419     OLDY=Y
420     Y=Y+DT
421     S=S+DS
422     IF(E .GT. 0.)GO TO 54
423     FLAG=1
424     IF(E .EQ. 0.)ICEASE=1
425 C
426 C:::::THIS STOPPING CRITERIA IS BASED ON VELOCITY GOING TO ZERO.
427 C
428     54 IF(FLAG .EQ. 1)GO TO 59
429     55 CONTINUE
430     OLDSIN=SINTH
431     OLDCOS=COSTH
432     CALL SDERIV(S,E,R)
433     IF(TRAPPD .EQ. 1 .OR. BLOC .GT. 0)GO TO 57
434     IF(%.GT. 0.)GO TO 56
435     RAT=R/(RO-R)
436     E13TRP=E13+(E13-E130)*RAT
437     ZTRAP=DEPTH-(Z+ZINCR*RAT)*DIA
438     SHTRAP=.245*(S+DS*RAT)*E13TRP
439     SHTRA=1.77*SHTRAP
440     PDEPTH=100.-(ZTRAP/DEPTH*100.)
441     FINALD=.308*(S+DS*RAT)*DIA
442     FINALT=OLDY+RAT*(Y-OLDY)
443     TRAPPD=1
444     NTRAP=N
445     GO TO 57
446     56 ROLD=RO
447     RO=R
448     E23=E130
449     E130=E13
450     57 N=N+1
451 C
452 C:::::ARE PLUMES FROM ADJACENT PORTS MERGING?
453 C
454     BOLD=BNEW
455     BNEW=.308*S*DIA

```

PROGRAM UPLUME

```

456      IF(BLOC .GT. 0)GO TO 58
457      IF(IMR .GT. 0)GO TO 80
458      IF(BNEW .LT. ASPAC)GO TO 80
459      NMERGE=N-1
460      RAT2=(SPACE-BOLD)/(BNEW-BOLD)
461      ZZMR=DIA*(Z-ZLAST)*RAT2
462      EMR=ZZMR+ZLAST*DIA
463      ZMR=DEPTH-EMR
464      XMR=DIA*(OLDX+RAT2*(X-OLDX))
465      TMR=OLDT+RAT2*(T-OLDT)
466      BMR=ASPAC
467      SINMR=OLDSIN+RAT2*(SINTH-OLDSIN)
468      COSMR=OLDCOS+RAT2*(COSTH-OLDCOS)
469      THETAM=ATAN2(SINMR,COSTH)*57.2958
470      SNEW=S-DS*(1.-RAT2)
471      IF(ANGLE .EQ. 90.)SMR=EMR
472      IF(ANGLE .NE. 90.)SMR=SNEW*DIA
473      CLMR=.245*SNEW*(E13+RAT2*(E13-E23))
474      DILMR=1.77*CLMR
475      IF(TRAPPD .EQ. 0)IMR=1
476      IF(TRAPPD .EQ. 1 .AND. NTRAP .NE. NMERGE)IMR=2
477      IF(TRAPPD .EQ. 1 .AND. NTRAP .EQ. NMERGE)IMR=3
478      IF(IPO .EQ. 2)GO TO 80
479      IF(IMR .EQ. 2)WRITE(10,213)
480      213 FORMAT(1X,' PLUMES MERGED')
481      IF(IMR .EQ. 1)WRITE(10,214)
482      214 FORMAT(1X,' PLUMES MERGED, PARAMETERS AT THAT TIME WERE:')
483      IF(IPO .EQ. 0 .AND. IMR .EQ. 1)WRITE(10,220)
484      *TMR,SMR,XMR,ZMR,BMR,EMR,THETAM,DILMR
485      IF(IPO .EQ. 1 .AND. IMR .EQ. 1)WRITE(10,220)
486      *TMR,SMR,XMR,ZMR,BMR,EMR,THETAM,CLMR,DILMR
487      IF(IMR .NE. 3)WRITE(10,215)
488      215 FORMAT(1X,' FOLLOWING CALCULATIONS DO NOT ACCOUNT FOR MERGING,
489      * A SINGLE PLUME IS ASSUMED.')
490      IF(IMR .EQ. 3)WRITE(10,216)
491      216 FORMAT(1X,' PLUMES MERGED AND TRAPPED AT THE SAME TIME.')
492      C
493      C::::HAS BLOCKING ZONE (90 PERCENT OF DEPTH) BEEN REACHED?
494      C
495      80 CONTINUE
496      IF(TRAPPD .EQ. 1 .OR. BLOC .GT. 0)GO TO 58
497      IF(Z .LT. ZRMAX)GO TO 58
498      NBLOC=N-1
499      RAT3=(ZRMAX-ZLAST)/(Z-ZLAST)
500      EBLOC=ZRMAX*DIA
501      ZBLOC=DEPTH-ZRMAX*DIA
502      XBLOC=DIA*(OLDX+RAT3*(X-OLDX))
503      TBLOC=OLDT+RAT3*(T-OLDT)
504      BBLOC=BNEW
505      SINBLO=OLDSIN+RAT3*(SINTH-OLDSIN)
506      COSBLO=OLDCOS+RAT3*(COSTH-OLDCOS)
507      THETAB=ATAN2(SINBLO,COSBLO)*57.2958
508      SNEW=S-DS*(1.-RAT3)
509      IF(ANGLE .EQ. 90.)SBLOC=EBLOC
510      IF(ANGLE .NE. 90.)SBLOC=SNEW*DIA
511      CLBLOC=.245*SNEW*(E13+RAT3*(E13-E23))
512      DBLOC=1.77*CLBLOC
513      IF(IMR .EQ. 0)BLOC=1
514      IF(IMR .EQ. 1 .AND. NMERGE .LT. NBLOC)BLOC=2
515      IF(IMR .EQ. 1 .AND. NMERGE .EQ. NBLOC)BLOC=3
516      IF(IPO .EQ. 2)GO TO 59
517      IF(BLOC .EQ. 2)WRITE(10,249)
518      249 FORMAT(1X,' BLOCKING HAS BEGUN')
519      IF(BLOC .EQ. 1)WRITE(10,250)
520      250 FORMAT(1X,' BLOCKING ZONE REACHED, PARAMETERS AT THAT TIME WERE:')

```

PROGRAM UPLUNE

```

521      IF(IPO .EQ. 0 .AND. BLOC .EQ. 1)WRITE(IO,220)
522      *TBLOC,SBLOC,XBLOC,ZBLOC,BBLOC,EBLOC,THETAB,DBLOC
523      IF(IPO .EQ. 1 .AND. BLOC .EQ. 1)WRITE(IO,220)
524      *TBLOC,SBLOC,XBLOC,ZBLOC,BBLOC,EBLOC,THETAB,CLBLOC,DBLOC
525      IF(BLOC .EQ. 3)WRITE(IO,251)
526      251 FORMAT(1X,' BLOCKING ZONE REACHED AND PLUMES MERGED
527      * AT THE SAME TIME.')
528      IF(IMR .EQ. 1 .OR. BLOC .EQ. 1)GO TO 51
529      C
530      C:::::IS IT TIME TO PRINT A LINE OF DATA?
531      C
532      58      IF(N-(N/NPO)*NPO .NE. 0)GO TO 51
533      59      XP=X+DIA
534      ELEV=Z+DIA
535      ZP=DEPTH-ELEV
536      SP=S+DIA
537      B=.308*SP
538      DILN=.245*S+E13
539      DILNA=1.77*DILN
540      THETA=ATAN2(SINTH,COSTH)*57.2958
541      IF(IPO .EQ. 2)GO TO 61
542      IF(TRAPPD .EQ. 0 .AND. BLOC .EQ. 0)GO TO 60
543      WRITE(IO,217)SP,XP,ZP,ELEV,THETA
544      217 FORMAT(10X,3F10.2,10X,F10.2,F10.1)
545      GO TO 61
546      60      IF(IPO .EQ. 1)GO TO 64
547      WRITE(IO,219)T,SP,XP,ZP,B,ELEV,THETA,DILNA
548      219 FORMAT(6F10.2,F10.1,F10.2)
549      GO TO 61
550      64      WRITE(IO,220)T,SP,XP,ZP,B,ELEV,THETA,DILN,DILNA
551      220 FORMAT(6F10.2,F10.1,2F10.2)
552      C
553      C:::::PRINT RESULTS.
554      C
555      61      IF(FLAG .EQ. 0)GO TO 51
556      IF(IPO .NE. 1)GO TO 62
557      WRITE(IO,221)
558      IF(TRAPPD .EQ. 1 .AND. IMR .NE. 3)WRITE(IO,222)ZTRAP,
559      *SMTRAP,SMTRA,FINALT,FINALB
560      IF(IMR .EQ. 3)WRITE(IO,218)ZTRAP,SMTRAP,SMTRA
561      IF(TRAPPD .EQ. 0)WRITE(IO,224)
562      IF(TRAPPD .EQ. 1)WRITE(IO,223)PDEPTH
563      IF(BLOC .GT. 0)GO TO 81
564      GO TO 63
565      62      IF(ICEASE .EQ. 1)WRITE(IO,240)
566      IF(ICEASE .EQ. 2)WRITE(IO,241)
567      IF(ICEASE .EQ. 3)WRITE(IO,242)
568      IF(IMR .EQ. 1 .AND. IPO .EQ. 2)WRITE(IO,243)DILMR,ZMR
569      IF(BLOC .GT. 0)GO TO 81
570      IF(IMR .EQ. 3)WRITE(IO,248)ZTRAP,SMTRA
571      IF(IMR .EQ. 2)WRITE(IO,244)ZMR
572      IF(TRAPPD .EQ. 1 .AND. IMR .NE. 3)WRITE(IO,245)ZTRAP,SMTRA,
573      *FINALT,FINALB
574      IF(TRAPPD .EQ. 0 .AND. IMR .LT. 2)WRITE(IO,246)DILNA
575      KPAGE=INTER
576      GO TO 63
577      81      CONTINUE
578      WRITE(IO,252)
579      IF(BLOC .EQ. 3)WRITE(IO,253)
580      WRITE(IO,254)ZBLOC,DBLOC
581      IF(BLOC .EQ. 2)WRITE(IO,255)
582      252 FORMAT(1X,' BLOCKING ZONE REACHED')
583      253 FORMAT(1X,' AND MERGING OCCURRED')
584      255 FORMAT(1X,' AFTER MERGING OCCURRED')
585      254 FORMAT(1X,' AT ',F7.2,' M BELOW WATER SURFACE,

```

PROGRAM UPLUME

```

586      * WITH AVE. DILUTION OF ',F6.1)
587 63 CONTINUE
588      IF(INTER .EQ. 0)GO TO 85
589      IF(ICEASE .EQ. 1)WRITE(IT,240)
590      IF(ICEASE .EQ. 2)WRITE(IT,241)
591      IF(ICEASE .EQ. 3)WRITE(IT,242)
592      IF(IMR .EQ. 1 .AND. IPO .EQ. 2)WRITE(IT,243)DILMR,ZMR
593      IF(BLOC .GT. 0)GO TO 82
594      IF(TRAPPD .EQ. 1 .AND. IMR .NE. 3)WRITE(IT,245)ZTRAP,SMTRA,
595      *FINALT,FINALB
596      IF(IMR .EQ. 2)WRITE(IT,244)ZMR
597      IF(IMR .EQ. 3)WRITE(IT,248)ZTRAP,SMTRA
598      IF(TRAPPD .EQ. 0 .AND. IMR .LT. 2)WRITE(IT,246)DILNA
599      GO TO 3
600 82 CONTINUE
601      WRITE(IT,252)
602      IF(BLOC .EQ. 3)WRITE(IT,253)
603      WRITE(IT,254)ZBLOC,DBLOC
604      IF(BLOC .EQ. 2)WRITE(IT,255)
605      GO TO 3
606 C
607 218 FORMAT(1X,' MERGING AND TRAPPING LEVEL IS: ',F7.2,' M BELOW
608      * WATER SURFACE'/1X,' WITH CL DILUTION OF ',F6.1,' AND
609      * AVE. DILUTION OF ',F6.1)
610 221 FORMAT(1X,' LAST LINE ABOVE IS FOR MAXIMUM HEIGHT OF RISE.')
611 222 FORMAT(1X,' TRAPPING LEVEL IS',F7.2,
612      * ' M WITH CL DILUTION OF',F7.1,' AND AVE. DILUTION OF',F7.1/
613      * 1X,' TIME TO TRAP:',F6.2,' SEC. PLUME DIA AT THE TRAPPING
614      * LEVEL:',F6.2,' M')
615 223 FORMAT(1X,' HEIGHT OF RISE=',F5.1,' PERCENT OF DEPTH')
616 224 FORMAT(1X,' TRAPPING LEVEL IS NOT REACHED')
617 240 FORMAT(/1X,' COMPUTATIONS CEASE: PLUME VELOCITY IS ZERO')
618 241 FORMAT(/1X,' COMPUTATIONS CEASE: PLUME SURFACES')
619 242 FORMAT(/1X,' COMPUTATIONS CEASE: PLUME TRAJECTORY IS',
620      * ' HORIZONTAL')
621 243 FORMAT(1X,' NOTE: AVERAGE DILUTION WAS',F6.1,' WHEN PLUMES'/
622      * 1X,' MERGED AT',F7.2,' M BELOW THE WATER SURFACE.'/
623      * 1X,' TRAPPING LEVEL NOT YET REACHED. AVE. DILUTION'/
624      * 1X,' SHOWN BELOW DOES NOT ACCOUNT FOR MERGING.')
625 244 FORMAT(1X,' PLUMES MERGED AT DEPTH',F7.2,' M, AFTER TRAPPING
626      * LEVEL REACHED.')
627 245 FORMAT(/1X,' TRAPPING LEVEL =',F7.2,' M BELOW WATER SURFACE.'/
628      * 1X,' AVERAGE DILUTION =',F6.1/
629      * 1X,' TIME TO TRAP:',F6.2,' SEC. PLUME DIA AT THE TRAPPING
630      * LEVEL:',F6.2,' M')
631 246 FORMAT(/1X,' AVERAGE DILUTION =',F7.1)
632 248 FORMAT(/1X,' PLUMES MERGED AND TRAPPED AT THE SAME TIME',
633      * F7.2,' M BELOW'/1X,' WATER SURFACE WITH AVERAGE DILUTION
634      * = ',F5.1)
635 C
636 C:::::FIND NEXT STRATIFICATION AND RECOMPUTE LAST STEP IF
637 C      NECESSARY.
638 C
639 70 DS=DS*(ZLIM-ZLAST)/(Z-ZLAST)
640      CALL SDERIV(S,E,R)
641      CHGDEN=1
642      Z=ZLAST
643      GO TO 52
644 71 CHGDEN=0
645      IPTS=IPTS+1
646      IF(IPTS .GT. NM)GO TO 72
647      G=GG(IPTS)
648      ZLIM=ZD(IPTS)
649      GO TO 53
650 C

```

PROGRAM UPLUME

```

651      72  IF(IPO .EQ. 1)WRITE(IO,230)
652      230 FORMAT(1X,' PLUME HITS SURFACE'/
653      *1X,' HEIGHT OF RISE=100.0 PERCENT OF DEPTH')
654      FLAG=1
655      ICEASE=2
656      GO TO 53
657      END
658  C
659      S'BROUTINE RPLUME(K,LIMIT)
660  C
661  C:::::READ UNIVERSAL DATA FILE AND CONVERT DATA FOR USE IN PROGRAM UPLUME
662  C
663      COMMON/AVAR/DP(30),SA(30),TA(30),RHO(30),DENPP(30)
664      COMMON/VAR/INTER,IDFP,ICUTOP,IPSF,IPI,IPO,ANGLE,Q,DIA,
665      *DEPTH,RHOJ,FN,PS,SPACE,SI,TI,NPTS,RFDD,UO,NQ
666      COMMON/SAV/II(5)
667      COMMON/COUNT/NMO,NW
668      COMMON/INOUT/IN,IO,IT
669      REAL*8 TITLE(10),TITLI(10),AA(10)
670      BYTE IFILE(20),OFILE(20)
671      COMMON/TVAR/TITLE,TITLI,IFILE
672      DATA IFILE(20)/0/,OFILE(20)/0/
673      IF(K .EQ. 2)GO TO 40
674      READ(IN,200,END=99,ERR=999)(TITLE(I),I=1,10)
675      200 FORMAT(10A8)
676      READ(IN,201,ERR=999)INTER,IDFP,ICUTOP,(II(J),J=1,5)
677      201 FORMAT(8I2)
678      READ(IN,202,ERR=999)QT,NP,PDIA,VANG,PDEP
679      202 FORMAT(F10.0,I10,3F10.0)
680      READ(IN,203,ERR=999)UW,HANG,SPACE
681      203 FORMAT(3F10.0)
682  C
683  C:::::CHANGE PROGRAMMED VARIABLE (PS)? ALL *X* VARIABLES ARE IGNORED.
684  C
685      IF(ICUTOP .EQ. 1)READ(IN,213,ERR=999)AX,IX1,IX2,IX3,IX4,IX5,
686      *PS,RX,DX,HX,EX,AAX,IX6,IX7
687      213 FORMAT(F5.0,2I5,3I2,6F5.0,2I5)
688  C
689      READ(IN,204,ERR=999)NPTS,S,T
690      204 FORMAT(I10,2F10.0)
691      READ(IN,205,ERR=999)((DP(I),SA(I),TA(I)),I=1,NPTS)
692      205 FORMAT(3F10.0)
693      IPI=II(1)
694      IPO=IPI
695      Q=QT
696      ANGLE=VANG
697      DIA=PDIA
698      DEPTH=PDEP
699      SI=S
700      TI=T
701      FN=NP
702      IF(ICUTOP .EQ. 0)GO TO 38
703      IPSF=1
704      IF(PS)38,38,39
705      38 PS=3.
706      IPSF=0
707      50 CONTINUE
708      39 RETURN
709      40 NW=NPTS+5+ICUTOP
710      IF(LIMIT .EQ. 0 .AND. IDFP .EQ. 1)WRITE(IO,403)
711      403 FORMAT(1H1)
712      IF(IDFP .EQ. 0)GO TO 50
713      WRITE(IO,400)(IFILE(IX),IX=1,19)
714      400 FORMAT(1X,' UNIVERSAL DATA FILE: ',19A1/)
715      IF(INTER .EQ. 1)WRITE(IO,10)

```

PROGRAM UPLUME

```

716 10 FORMAT(
717 *1X,' *****'/
718 *1X,' * NOTE, THIS IS THE ORIGINAL FILE. */
719 *1X,' * IT DOES NOT REFLECT CHANGES MADE INTERACTIVELY. */
720 *1X,' * THOSE CHANGES ARE SHOWN IN THE OUTPUT HEADING. */
721 *1X,' *****'/)
722 REWIND IN
723 IF(NWO .EQ. 0) GO TO 41
724 DO 1001 I=1,NWO
725 READ(IN,401)(AA(J),J=1,10)
726 1001 CONTINUE
727 41 DO 1000 I=1,NW
728 READ(IN,401)(AA(J),J=1,10)
729 401 FORMAT(10A8)
730 WRITE(IO,402)(AA(J),J=1,10)
731 402 FORMAT(1X,10A8)
732 1000 CONTINUE
733 GO TO 50
734 999 WRITE(IO,403)
735 WRITE(IO,998)(TITLE(IK),IK=1,10)
736 998 FORMAT(1X,' COMPUTATIONS CEASE FOR: '/2X,'CASE I.D. ',10A8/
737 *1X,' INPUT ERROR, CHECK DATA FILE')
738 IF(INTER .EQ. 1)WRITE(IT,998)(TITLE(IK),IK=1,10)
739 99 CALL EXIT
740 END
741 C
742 SUBROUTINE NPLUME(K,IFLAG)
743 C
744 C:::::PRINTS INPUT DATA FOR UPLUME IN NEW FORMAT.
745 C
746 COMMON/AVAR/DP(30),SA(30),TA(30),DENP(30),DENPP(30)
747 COMMON/VAR/INTER,IDFP,ICUTOP,IPSF,IPI,IPO,ANGLE,Q,DIA,
748 *DEPTH,RHOJ,FN,PS,SPACE,SI,TI,NPTS,RFDD,UO,NQ
749 REAL*8 TITLE(10),TITLI(10)
750 BYTE IFILE(20),OFILE(20)
751 COMMON/TVAR/TITLE,TITLI,IFILE
752 COMMON/INOUT/IN,IO,IT
753 DATA IFILE(20)/0/,OFILE(20)/0/
754 C
755 IF(K-2)1,2,3
756 1 IF(IPO .EQ. 2)GO TO 122
757 IF(IPSF .EQ. 0)WRITE(IO,110)PS
758 IF(IPSF .EQ. 1)WRITE(IO,111)PS
759 122 WRITE(IO,4)
760 IF(TI .NE. 0)WRITE(IO,123)TI
761 IF(TI .NE. 0)WRITE(IO,124)SI
762 WRITE(IO,125)RHOJ
763 WRITE(IO,133)UO
764 WRITE(IO,134)RFDD
765 IF(IFLAG .EQ. 1)WRITE(IO,131)
766 IF(IFLAG .EQ. 1)WRITE(IO,132)((DP(I),SA(I),TA(I),DENPP(I)),
767 *I=1,NPTS)
768 IF(IFLAG .EQ. 0)WRITE(IO,135)
769 IF(IFLAG .EQ. 0)WRITE(IO,136)((DP(I),SA(I)),I=1,NPTS)
770 2 WRITE(IO,200)
771 WRITE(IO,201)Q
772 WRITE(IO,202)NQ
773 WRITE(IO,203)DIA
774 IF(NQ .GT. 1)WRITE(IO,121)SPACE
775 IF(NQ .EQ. 1)WRITE(IO,126)
776 WRITE(IO,204)ANGLE
777 WRITE(IO,205)DEPTH
778 RETURN
779 3 CONTINUE
780 WRITE(IO,210)(TITLI(IK),IK=1,10)

```

PROGRAM UPLUME

```

781      WRITE(10,134)RFDD
782      GO TO 2
783  C
784      4 FORMAT(1X)
785      110 FORMAT(/2X,'PRINTOUT INTERVAL',19X,'=',F7.0,6X,'(DEFAULT)')
786      111 FORMAT(/2X,'PRINTOUT INTERVAL',19X,'=',F7.0)
787      121 FORMAT(2X,'PORT SPACING',24X,'=',F9.2,' M')
788      123 FORMAT(2X,'INITIAL TEMPERATURE OF THE PLUME',4X,'=',F9.2,4X,
789      *'DEGREES CENTIGRADE')
790      124 FORMAT(2X,'INITIAL SALINITY OF THE PLUME',7X,'=',F9.2,' PPT')
791      125 FORMAT(2X,'INITIAL DENSITY OF THE PLUME',8X,'=',F12.5,' G/CM3')
792      126 FORMAT(2X,'PORT SPACING',24X,'= 1000.0 M(DEFAULT)')
793      131 FORMAT(/3X,'DEPTH',4X,'SALIN',5X,'TEMP',3X,'SIGMAT',/4X,'(M)',
794      *5X,'(PPT)',5X,'(C)',/)
795      132 FORMAT(1X,F7.2,3F9.2)
796      133 FORMAT(2X,'DISCHARGE VELOCITY',18X,'=',F10.3,' M/S')
797      134 FORMAT(2X,'FROUDE NUMBER',23X,'=',F8.1)
798      135 FORMAT(/3X,'DEPTH',4X,'DENSITY')
799      136 FORMAT(1X,F7.2,F11.5)
800      200 FORMAT(1X)
801      201 FORMAT(2X,'TOTAL EFFLUENT FLOW',17X,'=',F11.4,' CMS')
802      202 FORMAT(2X,'NUMBER OF PORTS',21X,'=',I6)
803      203 FORMAT(2X,'PORT DIAMETER',23X,'=',F11.4,' M')
804      204 FORMAT(2X,'VERTICAL PORT ANGLE FROM HORIZONTAL =',F8.1,5X,
805      *'DEGREES')
806      205 FORMAT(2X,'PORT DEPTH',26X,'=',F9.2,' M')
807      210 FORMAT(/2X,70('X'))/2X,'RUN TITLE: ',10A8/)
808      END
809  C
810      SUBROUTINE LIMITS(KK,PDEP,VANG,T,S,NPTS,INTER,TITLE,TITLI,
811      *IEXIT,LIMIT)
812  C
813  C      THIS SUBROUTINE CHECKS LIMITS OF PORT DEPTH (GREATER THAN 0. AND
814  C      LESS THAN OR EQUAL TO PROFILE DEPTH), DISCHARGE ANGLE (EQUAL TO
815  C      OR GREATER THAN -5 DEG BUT LESS THAN OR EQUAL TO 90 DEG,
816  C      VERTICAL, AND IF NEGATIVE, THAT THERE IS AT LEAST ONE PROFILE
817  C      DATA POINT AT LEAST ONE METER BELOW THE PORT DEPTH). THAT THE
818  C      EFFLUENT DENSITY IS LESS THAN THE AMBIENT DENSITY AND THAT THERE
819  C      ARE AMBIENT PROFILE VALUES FOR THE SURFACE, IF INTER=1, THE PORT
820  C      DEPTH, ANGLE AND EFFLUENT DENSITY CAN BE CORRECTED INTERACTIVELY
821  C      BUT SURFACE DATA AND PROFILE DEPTH CORRECTIONS MUST BE MADE TO
822  C      THE DATA SET(S) AND REENTERED. IF INTER=0, ALL CORRECTION MUST BE
823  C      MADE TO THE DATA SET(S) AND REENTERED.
824  C
825      COMMON/AVAR/DP(30),SA(30),TA(30),RHO(30),DENPP(30)
826      COMMON/INOUT/IN,IO,IT
827      REAL*8 TITLE(10),TITLI(10)
828      LIMIT=0
829      IDEEP=0
830      IVANG=0
831      NEGANG=0
832      IRHO=0
833      INO=0
834      IEXIT=0
835      100 IF(PDEP .EQ. 0. .OR. DP(NPTS) .LT. PDEP)GO TO 1
836      10 IF(VANG .LT. 0. .OR. VANG .GT. 90.)GO TO 101
837      30 IF(T .EQ. 0.)RHOE=S
838      IF(T .NE. 0.)RHOE=1.+0.001*SIGMAT(S,T)
839      DO 31 J=2,NPTS
840      IF(PDEP .LE. DP(J))GO TO 32
841      31 CONTINUE
842      GO TO 40
843      32 JK=J
844      PRO=(PDEP-DP(JK-1))/(DP(JK)-DP(JK-1))
845      TT=TA(JK-1)+PRO*(TA(JK)-TA(JK-1))

```

PROGRAM UPLUME

```

846      SS=SA(JK-1)+PRO*(SA(JK)-SA(JK-1))
847      IF(TA(JK) .EQ. 0.)RHOA=SS
848      IF(TA(JK) .NE. 0.)RHOA=1.+0.001*SIGMAT(SS,TT)
849      IF(RHOA .GT. RHOE)GO TO 40
850      IRHO=1
851      LIMIT=1
852      IF(INTER .EQ. 0)GO TO 40
853 120 CONTINUE
854      WRITE(IT,33)RHOE,RHOA
855 33 FORMAT(/1X,' EFFLUENT DENSITY (' ,F7.5,' G/CM3) MUST BE'/
856      *1X,' LESS THAN THE AMBIENT DENSITY (' ,F7.5,' G/CM3)')
857      WRITE(IT,4)
858      READ(IT,5)NANS
859      IF(NANS .EQ. 'N')GO TO 40
860      IF(NANS .NE. 'Y')WRITE(IT,200)
861      IF(NANS .NE. 'Y')GO TO 120
862      IRHO=2
863      IF(T .EQ. 0.)GO TO 36
864      WRITE(IT,34)
865 34 FORMAT(/1X,' NEW EFFLUENT TEMP VALUE? ' ,S)
866      READ(IT,7)T
867      WRITE(IT,35)
868 35 FORMAT(1X,' NEW EFFLUENT SAL VALUE? ' ,S)
869      READ(IT,7)S
870      GO TO 30
871 36 WRITE(IT,37)
872 37 FORMAT(/1X,' NEW EFFLUENT DENSITY VALUE? ' ,S)
873      READ(IT,7)S
874      GO TO 30
875 101 IF(VANG .LT. -5. .OR. VANG .GT. 90.)GO TO 11
876 C
877 C:::::IF CONTROL IS NOT PASSED TO 11, VANG .GE. -5. BUT .LT. 0.
878 C      BECAUSE IT'S NEGATIVE, PROFILE DEPTH MUST BE AT LEAST ONE
879 C      METER DEEPER THAN THE PORT DEPTH.
880 C
881      IF(DP(NPTS)-PDEP .GE. 1.0)GO TO 30
882      NEGANG=1
883 C
884      1 IDEEP=1
885      LIMIT=1
886      IF(INTER .EQ. 0)GO TO 10
887 121 CONTINUE
888      IF(PDEP .EQ. 0.)WRITE(IT,2)
889      2 FORMAT(/1X,' ZERO DISCHARGE DEPTH NOT ALLOWED,
890      * CHANGE PORT DEPTH.')
891      IF(PDEP .GT. 0. .AND. DP(NPTS) .LT. PDEP)WRITE(IT,3)PDEP,DP(NPTS)
892      3 FORMAT(/1X,' PORT DEPTH (' ,F6.2,' M) IS DEEPER THAN'/
893      *1X,' THE PROFILE DEPTH (' ,F6.2,' M), CHANGE PORT DEPTH.')
894      IF(NEGANG .EQ. 1)WRITE(IT,21)PDEP,DP(NPTS)
895 21 FORMAT(/1X,' FOR A NEGATIVE DISCHARGE ANGLE, THE PORT DEPTH ('
896      * ,F8.4,' M) MUST BE AT'/1X,' LEAST ONE METER LESS THAN THE PROFILE
897      * DEPTH (' ,F8.4,' M), CHANGE PORT DEPTH.')
898      WRITE(IT,4)
899      4 FORMAT(/1X,' WANT TO MAKE CORRECTION? YES/NO ' ,S)
900      READ(IT,5)NANS
901      5 FORMAT(A1)
902      IF(NANS .EQ. 'N')GO TO 10
903      IF(NANS .NE. 'Y')WRITE(IT,200)
904      IF(NANS .NE. 'Y')GO TO 121
905      IDEEP=2
906      WRITE(IT,6)
907      6 FORMAT(1X,' NEW VALUE? ' ,S)
908      READ(IT,7)PDEP
909      7 FORMAT(F7.0)
910      GO TO 100

```

PROGRAM UPLUME

```

911      11 IVANG=1
912      LIMIT=1
913      IF(INTER .EQ. 0)GO TO 30
914      122 CONTINUE
915      WRITE(IT,12)VANG
916      12 FORMAT(/1X,' DISCHARGE ANGLE ('F6.1,') IS NOT IN THE
917      * RANGE OF -5 TO 90 DEG')
918      WRITE(IT,4)
919      READ(IT,5)NANS
920      IF(NANS .EQ. 'N')GO TO 30
921      IF(NANS .NE. 'Y')WRITE(IT,200)
922      IF(NANS .NE. 'Y')GO TO 122
923      IVANG=2
924      WRITE(IT,6)
925      READ(IT,7)VANG
926      GO TO 10
927      40 IF(DP(1) .EQ. 0.)GO TO 50
928      INO=1
929      LIMIT=1
930      IF(INTER .EQ. 0)GO TO 50
931      WRITE(IT,41)(TITLE(K),K=1,10)
932      41 FORMAT(1H1/1X,' COMPUTATIONS CEASE FOR'/1X,' CASE I.D. ',10A8)
933      WRITE(IT,44)(TITLI(K),K=1,10)
934      44 FORMAT(1X,' RUN TITLE: ',10A8)
935      WRITE(IT,42)
936      42 FORMAT(/1X,' NO SURFACE DATA FOR THE AMBIENT DENSITY/CURRENT
937      * PROFILE.'/1X,' WITHOUT THESE DATA, COMPUTATIONAL ERRORS
938      * MAY OCCUR.'/1X,' MAKE CORRECTIONS TO DATA SET(S) AND REENTER.')
```

```

939      WRITE(IT,43)
940      43 FORMAT(/1X,' GOING TO NEXT DATA SET IF THERE IS ONE.')
```

```

941      50 IF(LIMIT .EQ. 0)GO TO 70
942      IF(INTER .EQ. 1)GO TO 69
943      51 WRITE(IO,41)(TITLE(K),K=1,10)
944      IF(INTER .EQ. 0)GO TO 73
945      IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
946      WRITE(IT,41)(TITLE(K),K=1,10)
947      IF(KK .EQ. 2)WRITE(IT,44)(TITLI(K),K=1,10)
948      WRITE(IT,74)
949      74 FORMAT(/1X,' NECESSARY CORRECTIONS NOT MADE!')
```

```

950      WRITE(IT,43)
951      73 WRITE(IO,67)
952      67 FORMAT(/1X,' CORRECT THE FOLLOWING AND REENTER DATA.')
```

```

953      IEXIT=1
954      68 IF(IDEEP .EQ. 0)GO TO 52
955      IF(IDEEP .EQ. 1)WRITE(IO,56)PDEP
956      IF(IDEEP .EQ. 2)WRITE(IO,57)PDEP
957      52 IF(IVANG .EQ. 0)GO TO 54
958      IF(IVANG .EQ. 1)WRITE(IO,58)VANG
959      IF(IVANG .EQ. 2)WRITE(IO,59)VANG
960      54 IF(IRHO .EQ. 0)GO TO 55
961      IF(IRHO .EQ. 1)WRITE(IO,64)
962      IF(IRHO .EQ. 2)WRITE(IO,65)RHOE
963      55 IF(INO .EQ. 0)GO TO 75
964      WRITE(IO,42)
965      WRITE(IO,43)
966      GO TO 70
967      69 IF(IDEEP .EQ. 1 .OR. IVANG .EQ. 1 .OR. IRHO .EQ. 1
968      *.OR. INO .EQ. 1)GO TO 51
969      WRITE(IO,66)(TITLE(K),K=1,10)
970      IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
971      WRITE(IO,71)
972      GO TO 68
973      56 FORMAT(
974      *1X,' PORT DEPTH WAS ENTERED AS ',F6.2,' METERS:/'
975      *1X,' IF THE DISCHARGE ANGLE IS POSITIVE, THE PORT DEPTH MUST'/
```

PROGRAM UPLUME

```

976      *1X,' BE GREATER THAN ZERO BUT LESS THAN OR EQUAL TO THE PROFILE
977      * DEPTH.'/
978      *1X,' IF THE DISCHARGE ANGLE IS NEGATIVE, THE PORT DEPTH MUST'/
979      *1X,' BE AT LEAST ONE METER LESS THAN THE PROFILE DEPTH.')
980      57 FORMAT(1X,' PORT DEPTH CHANGED TO: ',F6.2,' M')
981      58 FORMAT(1X,' DISCHARGE ANGLE (',F6.1,')
982      * MUST BE .GE. -5 BUT .LE. 90 DEG')
983      59 FORMAT(1X,' DISCHARGE ANGLE CHANGED TO: ',F4.1,' DEG')
984      64 FORMAT(1X,' EFFLUENT DENSITY MUST BE .LT. AMBIENT
985      * DENSITY AT THE DISCHARGE DEPTH')
986      65 FORMAT(1X,' EFFLUENT DENSITY CHANGED TO: ',F7.5,' G/CM3')
987      66 FORMAT(1H1/1X,' COMPUTATIONS CONTINUE FOR'/1X,' CASE I.D. ',10A8)
988      71 FORMAT(1X,' CORRECTIONS WERE INTERACTIVELY MADE TO
989      * THE FOLLOWING: '/')
990      200 FORMAT(/1X,' *****YOU MADE A MISTAKE, TRY AGAIN*****')
991      75 IF(IEXIT .EQ. 1)WRITE(10,43)
992      70 RETURN
993      END
994  C
995      SUBROUTINE USORT(DP,X,Y,Z,NPTS)
996  C
997  C:::::THIS SUBROUTINE SORTS THE AMBIENT PROFILE ON DEPTH,
998  C      SHALLOW TO DEEP.
999  C
1000     DIMENSION DP(30),X(30),Y(30),Z(30)
1001     NESTED=NPTS
1002     L=NESTED-1
1003     DO 1000 M=1,L
1004         NESTED=NESTED-1
1005         DO 1000 I=1,NESTED
1006             IF(DP(I) .LE. DP(I+1))GO TO 1000
1007             DUMMY=DP(I)
1008             DP(I)=DP(I+1)
1009             DP(I+1)=DUMMY
1010             DUMMY=X(I)
1011             X(I)=X(I+1)
1012             X(I+1)=DUMMY
1013             DUMMY=Y(I)
1014             Y(I)=Y(I+1)
1015             Y(I+1)=DUMMY
1016             DUMMY=Z(I)
1017             Z(I)=Z(I+1)
1018             Z(I+1)=DUMMY
1019     1000 CONTINUE
1020     RETURN
1021     END
1022  C
1023     FUNCTION SIGMAT(SAL,T)
1024  C
1025  C:::::THIS FUNCTION COMPUTES SIGMA T GIVEN TEMPERATURE AND SALINITY
1026  C      AFTER THE METHOD IN H.O. PUB. 615, TABLES FOR SEA WATER
1027  C      DENSITY, U.S. NAVY HYDROGRAPHIC OFFICE, WASHINGTON, D.C.
1028  C
1029     SIGO=(((6.8E-6*SAL)-4.82E-4)*SAL+.8149)*SAL-0.093
1030     B=1.E-6*T*((.01667*T-.8164)*T+18.03)
1031     A=.001*T*((.0010843*T-.09818)*T+4.7867)
1032     SUMT=(T-3.98)*(T-3.98)*(T+283.)/(503.57*(T+67.26))
1033     SIGMAT=(SIGO+.1324)*(1.-A+B*(SIGO-.1324))-SUMT
1034     RETURN
1035     END
1036  C
1037     SUBROUTINE SDERIV(S,E,R)
1038  C
1039  C:::::THIS SUBROUTINE IS PART OF THE 4TH ORDER
1040  C      RUNGE-KUTTA INTEGRATION.

```

PROGRAM UPLUME

```

1041 C
1042 COMMON G,FK,FH,COSTH,SINTH,COSTHE,DS,C1,C2,E13,FLAG,GRAV
1043 *,ICEASE
1044 COMMON CUBRT,TWOTH
1045 INTEGER FLAG
1046 E13=0.
1047 IF(E .LE. 0.)GO TO 3
1048 E13=E**CUBRT
1049 COSTH=COSTHE+C1/(E13+E13)
1050 IF(COSTH .LT. 1.)GO TO 1
1051 C
1052 C:::::THIS STOPPING CRITERIA IS BASED ON PLUME BECOMING HORIZONTAL
1053 C AGAIN.
1054 C
1055 FLAG=1
1056 ICEASE=3
1057 SINTH=0.
1058 COSTH=1.
1059 GO TO 3
1060 1 SINTH=SQRT(1.-COSTH*COSTH)
1061 3 FK=C2*S*R*SINTH*DS
1062 FH=.109*S*E13*G*DS*SINTH
1063 RETURN
1064 END

```

SECTION 3
UOUTPLM Listing

PROGRAM UOUTPLM

```

1  PROGRAM UOUTP
2  C
3  C   THIS IS PROGRAM UOUTPLM.  IN THE ABOVE, PROGRAM NAME MUST
4  C   NOT BE MORE THAN SIX CHARACTERS.
5  C
6  C   OUTPLM MODIFIED TO USE UNIVERSAL DATA FILE WITH OPTIONS
7  C
8  C   UP-TO-DATE AS OF AUGUST 1985 (NO CHANGES SINCE 7-31-85).
9  C   JULY 31, 1985 REMOVED/MODIFIED ALL STATEMENTS INVOLVING ICEASE=1
10 C   (NO DENSITY DATA FOR A GIVEN DEPTH) AS THAT CONDITION IS
11 C   DETECTED IN SUBROUTINE LIMITS AND EITHER CORRECTED INTERACTIVELY
12 C   OR THE PROGRAM GOES TO THE NEXT DATA SET.
13 C   JUNE 18, 1985 REVISED INTERACTIVE DATA INPUT SO USER MAY CORRECT
14 C   ERRORS BEFORE RERUNNING THE PROGRAM.
15 C   MARCH 27, 1985 CHANGE ANGLE OF DISCHARGE LIMITS FROM 0 TO 90 DEG
16 C   TO -5 TO 90 DEG. AND THAT DZ CAN BE NEGATIVE IF Z .GE. PDEP
17 C   MARCH 19, 1985 RELOCATED THE INTERACTIVE RECYCLE STATEMENT
18 C   (3 WRITE) TO 3 CONTINUE, THE FIRST STATEMENT UNDER INTERACTIVE
19 C   CONTROL. ADDED STATEMENTS IN SUBROUTINE LIMITS TO DETECT
20 C   ERRONEOUS RESPONSES TO PROGRAM PROMPTS I.E. .NE. TO YES OR NO.
21 C   FEB 7, 1985. REVISED THE CALCULATION OF
22 C   AK AND EINS IN THE MAIN COMPUTATIONAL SCHEME.
23 C
24 C   COMMON/WRITE/DPW(30),SAW(30),TAW(30),DENPW(30),DENPPW(30)
25 C   COMMON/AVAR/DP(30),SA(30),TA(30),DENP(30),DENPP(30)
26 C   COMMON/VAR/INTER,IDFP,ICUTOP,IOI,IOO,IHF,IEF,IAF,IIF,IIRF,QT,NP,
27 C   *PDIA,VANG,PDEP,UW,IR,HANG,SPACE,H,E,A,ITERD,NPTS,T,S,DEN,FR
28 C   COMMON/COUNT/NWO,NW
29 C   COMMON/INOUT/IN,IO,IT
30 C   REAL*8 TITLE(10),TITLI(10)
31 C   BYTE IFILE(20),OFILE(20)
32 C   COMMON/TVAR/TITLE,TITLI,IFILE
33 C   DATA IFILE(20)/O/,OFILE(20)/O/
34 C
35 C   C:::::1 INITIALIZE CONSTANTS AND VARIABLES.
36 C
37 C   IN THIS PROGRAM (PDP SYSTEM - FORTRAN IV-PLUS),
38 C   LOGICAL UNIT IN IS THE INPUT FILE.
39 C   "      "      IO IS THE OUTPUT FILE.
40 C   "      "      IT IS THE USER TERMINAL.
41 C
42 C   IN=1
43 C   IO=2
44 C   IT=5
45 C
46 C   NWO=0
47 C   NW=0
48 C   G=9.807
49 C   ZERO=0.
50 C   TWO=2.
51 C   PI=3.1416
52 C   TORAD=PI/180.
53 C
54 C   OPEN FILES
55 C
56 C   WRITE(IT,554)
57 C   554 FORMAT(1X,' *****PROGRAM UOUTPLM, AUGUST 1985*****')
58 C   WRITE(IT,555)
59 C   555 FORMAT(1X,' ENTER UDF NAME: ',S)
60 C   READ(IT,556)(IFILE(I),I=1,19)
61 C   556 FORMAT(19A1)
62 C   WRITE(IT,557)
63 C   557 FORMAT(1X,' ENTER OUTPUT FILE NAME: ',S)
64 C   READ(IT,556)(OFILE(I),I=1,19)
65 C   OPEN(UNIT=IN,NAME=IFILE,TYPE='OLD')

```

PROGRAM UOUTPLM

```

66      OPEN(UNIT=10,NAME=OFILE,TYPE='NEW')
67 C
68 C      READ DATA FILE
69 C
70      85 NWO=NWO+NW
71      NCW=0
72      LIMIT=0
73      KPAGE=0
74      CALL ROUTP(1,LIMIT)
75      XSCALE=0.
76      ZSCALE=0.
77      CALL USORT(DP,SA,TA,DENP,NPTS)
78      DO 1003 J=1,NPTS
79          DPW(J)=DP(J)
80          SAW(J)=SA(J)
81          TAW(J)=TA(J)
82      1003 CONTINUE
83      CALL LIMITS(1,PDEP,VANG,T,S,NPTS,INTER,TITLE,TITLI,
84      *IEXIT,LIMIT)
85      IF(IEXIT .EQ. 0)GO TO 38
86      CALL ROUTP(2,LIMIT)
87      GO TO 85
88      38 IFLAG=1
89      IF(TA(NPTS) .EQ. 0.)IFLAG=0
90      KFLAG=1
91      IF(T .EQ. 0.)KFLAG=0
92      IF(IFLAG .EQ. 0)GO TO 1
93      DO 1000 I=1,NPTS
94          DENP(I)=1000.+SIGMAT(SA(I),TA(I))
95          DENPW(I)=DENP(I)
96      1000 CONTINUE
97      GO TO 2
98      1 DO 1001 I=1,NPTS
99          DENP(I)=1000.*SA(I)
100          DENPW(I)=DENP(I)
101      1001 CONTINUE
102      2 DO 1002 I=1,NPTS
103          DENPP(I)=DENP(I)-1000.
104          DENPPW(I)=DENPP(I)
105      1002 CONTINUE
106 C
107 C      IF ONLY ONE DISCHARGE PORT, PORT SPACING IS SET
108 C      EQUAL TO 1000. TO PREVENT PLUMES MERGING FLAG.
109 C
110      IF(NP .EQ. 1)SPACE=1000.
111      IF(INTER .EQ. 0)GO TO 19
112 C
113 C      INTERACTIVE CONTROL
114 C
115      3 CONTINUE
116      IF(NCW .EQ. 0)TQ=T
117      IF(NCW .EQ. 0)SQ=S
118      IF(NCW .EQ. 0)GO TO 11
119      WRITE(IT,10)(TITLE(IK),IK=1,10)
120      10 FORMAT(/1X,' DO YOU WISH TO CHANGE ANY INPUT VARIABLES AND RERUN?'
121      */1X,' CASE I.D. ',10A8/1X,' YES TO RERUN OR NO FOR NEXT CASE: ',3)
122      READ(IT,9)ANS
123      9 FORMAT(A1)
124      IF(ANS .EQ. 'Y')GO TO 11
125      IF(ANS .EQ. 'N')GO TO 85
126      WRITE(IT,12)
127      12 FORMAT(/1X,' ***** YOU MADE A MISTAKE, TRY AGAIN *****')
128      GO TO 3
129      11 CONTINUE
130      IF(NCW .EQ. 0)WRITE(IT,7)(TITLE(IK),IK=1,10)

```

PROGRAM UOUTPLM

```

131      7 FORMAT(1X,' CASE IDENTIFICATION: '/
132      *2X,10A8)
133      WRITE(IT,13)
134      13 FORMAT(1X,' WHAT IS THE TITLE OF THIS INTERACTIVE RUN? '/
135      *2X,S)
136      READ(IT,6)(TITLI(IK),IK=1,10)
137      6 FORMAT(10A8)
138      IF(NCW .EQ. 0) GO TO 131
139      14 CONTINUE
140      WRITE(IT,110)QT,NP,PDIA,SPACE,VANG,PDEP,UW
141      110 FORMAT(1X,' YOUR PRESENT INPUT VARIABLES ARE: '//
142      *1X,' 1. EFFLUENT FLOW (CMS)          QT =',F10.4/
143      *1X,' 2. NUMBER OF PORTS              NP =',I5/
144      *1X,' 3. PORT DIAMETER (M)           PDIA =',F10.4/
145      *1X,' 4. PORT SPACING (M)            SPACE =',F9.3/
146      *1X,' 5. VERTICAL ANGLE W/HORIZ (DEG) VANG =',F9.3/
147      *1X,' 6. PORT DEPTH (M)              PDEP =',F8.2/
148      *1X,' 7. AMBIENT CURRENT SPEED (M/SEC) UW =',F10.4)
149      IF(TQ .EQ. 0.)WRITE(IT,130)SQ
150      130 FORMAT(
151      *1X,' 8. EFFLUENT DENSITY (G/CM3)      S =',F11.5)
152      IF(TQ .NE. 0.)WRITE(IT,112)SQ,TQ
153      112 FORMAT(
154      *1X,' 8. EFFLUENT SAL (0/00)          S =',F8.2/
155      *1X,' 9. EFFLUENT TEMP (DEG C)        T =',F8.2)
156      WRITE(IT,132)
157      132 FORMAT(1X,' ENTER THE NUMBER OF THE VARIABLE
158      * YOU WISH TO CHANGE. ',S)
159      READ(IT,115)NUMBER
160      115 FORMAT(I1)
161      IF(TQ .EQ. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 8)GO TO 15
162      IF(TQ .NE. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 9)GO TO 15
163      WRITE(IT,12)
164      GO TO 14
165      15 CONTINUE
166      WRITE(IT,16)
167      16 FORMAT(1X,' WHAT IS THE VALUE OF THIS VARIABLE? ',S)
168      READ(IT,18)VALUE
169      18 FORMAT(F10.0)
170      IF(NUMBER .EQ. 1)QT=VALUE
171      IF(NUMBER .EQ. 2)NP=VALUE
172      IF(NUMBER .EQ. 3)PDIA=VALUE
173      IF(NUMBER .EQ. 4)SPACE=VALUE
174      IF(NP .EQ. 1.)SPACE=1000.
175      IF(NUMBER .EQ. 5)VANG=VALUE
176      IF(NUMBER .EQ. 6)PDEP=VALUE
177      IF(NUMBER .EQ. 7)UW=VALUE
178      IF(NUMBER .EQ. 8)SQ=VALUE
179      IF(NUMBER .EQ. 9)TQ=VALUE
180      133 CONTINUE
181      WRITE(IT,17)
182      17 FORMAT(1X,' DO YOU WISH TO CHANGE ANOTHER VARIABLE?
183      * YES OR NO: ',S)
184      READ(IT,9)ANS
185      IF(ANS .EQ. 'N')GO TO 8
186      IF(ANS .EQ. 'Y')GO TO 14
187      WRITE(IT,12)
188      GO TO 133
189      8 CONTINUE
190      T=TQ
191      S=SQ
192      CALL LIMITS(2,PDEP,VANG,T,S,NPTS,INTER,TITLE,TITLI,IEXIT,LIMIT)
193      IF(IEXIT .EQ. 0)GO TO 131
194      CALL ROUTP(2,LIMIT)
195      GO TO 85

```

PROGRAM UOUTPLM

```

196 131 NCW=1
197 C
198 C   SETUP, AND PRINT INPUT DATA
199 C
200 19 CONTINUE
201 KK=0
202 DO 120 K=1,NPTS
203 DP(K)=DPW(K)
204 SA(K)=SAW(K)
205 TA(K)=TAW(K)
206 DENP(K)=DENPW(K)
207 IF(KK .GT. 0) GO TO 120
208 IF(DP(K) .GE. PDEP) KK=K
209 120 CONTINUE
210 PRODIS=(PDEP-DP(KK-1))/(DP(KK)-DP(KK-1))
211 IF(IFLAG .EQ. 1) GO TO 123
212 DENA=DENP(KK-1)+PRODIS*(DENP(KK)-DENP(KK-1))
213 GO TO 124
214 123 TAA=TA(KK-1)+PRODIS*(TA(KK)-TA(KK-1))
215 SAA=SA(KK-1)+PRODIS*(SA(KK)-SA(KK-1))
216 DENA=1000.+SIGMAT(SAA,TAA)
217 124 CONTINUE
218 IF(T .EQ. 0.) DEN=S*1000.
219 IF(T .NE. 0.) DEN=1000.+SIGMAT(S,T)
220 RHO=DEN
221 DENDIF=DENA-DEN
222 DENOLD=DENDIF
223 B=PDIA*0.5
224 SPAC2=SPACE*0.5
225 BSNEW=B-SPAC2
226 VEL=(QT/NP)/(PI*B*B)
227 U=VEL*COS(VANG*TORAD)
228 V=VEL*SIN(VANG*TORAD)
229 IF(VANG .EQ. 90.) U=0.
230 IF(VANG .EQ. 0.) V=0.
231 V1=VEL
232 FR=VEL/SQRT(DENDIF/DEN*TWO*B*G)
233 ZWEI=ZERO
234 BSAVE=ZERO
235 X=ZERO
236 DB=1.
237 DD=1.
238 DT=1.
239 Z=PDEP
240 IF(KFLAG*IFLAG .EQ. 1) DEL=T-TA(NPTS)
241 DIL=1.
242 PHO=PI*B*B*H*DEN
243 PH=PHO
244 Q=PH/DEN*VEL/H
245 AK=9999.
246 IF(ABS(UW) .NE. ZERO) AK=VEL/UW
247 UO=U
248 ITRAP=0
249 ILAP=0
250 ILAPP=0
251 JPH=0
252 JTL=0
253 IF(IOI .EQ. 1) GO TO 22
254 IF(KPAGE .EQ. 1) GO TO 20
255 CALL ROUTP(2,LIMIT)
256 WRITE(IO,102)
257 102 FORMAT(1H1,1X,'UOUTPLM VERSION 1.0 AUGUST 1985 (BASED ON OS3'
258 *,' VERSION 2.3 5-16-79)'/,1X,4(18(' '),2X))
259 WRITE(IO,119) (IFILE(IK),IK=1,19)
260 119 FORMAT(1X,' UNIVERSAL DATA FILE: ',19A1)

```

PROGRAM UOUTPLM

```

261      WRITE(IO,111)(TITLE(IK),IK=1,10)
262 111  FORMAT(2X,'CASE I.D. ',10A8)
263      IF(INTER.EQ. 1)WRITE(IO,113)(TITLI(IK),IK=1,10)
264 113  FORMAT(2X,'RUN TITLE: ',10A8)
265      CALL NOUTP(1,IFLAG)
266      GO TO 23
267 20   IF(IOI.EQ. 2)GO TO 21
268      CALL ROUTP(2,LIMIT)
269      WRITE(IO,102)
270      WRITE(IO,119)(IFILE(IK),IK=1,19)
271      WRITE(IO,111)(TITLE(IK),IK=1,10)
272      IF(INTER.EQ. 1)WRITE(IO,113)(TITLI(IK),IK=1,10)
273      CALL NOUTP(1,IFLAG)
274      GO TO 23
275 21   CALL NOUTP(3,IFLAG)
276      GO TO 23
277 22   CALL ROUTP(2,LIMIT)
278      WRITE(IO,101)
279 101  FORMAT(1H1,1X,'.....
280      *OUTFALL BUOYANT (JET) PLUME IN FLOWING, STRATIFIED AMBIENT
281      *.....'/
282      *1X,' UOUTPLM VERSION 1.0 AUGUST 1985 (BASED ON OS3 VERSION 2.3
283      * 5-16-79)')
284      WRITE(IO,119)(IFILE(IK),IK=1,19)
285      WRITE(IO,111)(TITLE(I),I=1,10)
286      IF(INTER.EQ. 1)WRITE(IO,113)(TITLI(IK),IK=1,10)
287      WRITE(IO,100)E,A,ITERB,IR
288 100  FORMAT(1X,' E=',F5.2,' A=',F5.2,' ITERB=',I5,' IR=',I4)
289      IF(NP.EQ. 1)WRITE(IO,43)PDIA,VANG
290 43   FORMAT(1X,' PORT SPACING (M) = 1000.0 (DEFAULT),',2X,
291      *'PORT DIA (M) =',F7.4,' , PORT ANGLE (DEG) =',F5.1)
292      IF(NP.GT. 1)WRITE(IO,44)SPACE,PDIA,VANG
293 44   FORMAT(1X,' PORT SPACING (M) = ',F6.2,
294      *', PORT DIA (M) =',F7.4,' , PORT ANGLE (DEG) =',F5.1)
295      IF(IFLAG.EQ.1)GO TO 105
296      WRITE(IO,117)
297 117  FORMAT(1X,' AMBIENT STRATIFICATION DEPTH,M SIGMAT')
298      WRITE(IO,104)((DPW(I),DENPPW(I)),I=1,NPTS)
299 104  FORMAT(26X,F7.2,F8.2)
300      GO TO 23
301 105  WRITE(IO,107)
302 107  FORMAT(1X,' AMBIENT STRATIFICATION DEPTH,M SALIN
303      * TEMP,C SIGMAT')
304      WRITE(IO,103)((DPW(I),SAW(I),TAW(I),DENPPW(I)),I=1,NPTS)
305 103  FORMAT(26X,2F7.2,2F8.2)
306 23   KPAGE=INTER
307      IF(100.EQ. 1)WRITE(IO,200)AK,FR,Q,UW
308 200  FORMAT(1X,'.....
309      *' K FROUDE Q CURRENT'/1X,4(1PE9.2)//2X,
310      *'-----MODEL INPUT (LINE 1) AND MODEL OUTPUT-----'/
311      * Z B THICK MASS EINS ZWEI DILUTION
312      * DENDIFF HOR VEL VER VEL TOT VEL')
313      IF(100.EQ. 1.AND. KFLAG=IFLAG.EQ. 1)WRITE(IO,201)
314 201  FORMAT(1H1,110X,'TEMPIF')
315      IF(100.EQ. 0)WRITE(IO,210)
316 210  FORMAT(1X,7X,'X',8X,'Z',6X,'PLUME',5X,'DILU-',2X,'DENDIFF',3X,
317      *'HORIZ',5X,'VERT',4X,'TOTAL',1X,22X,'RADIUS',5X,'TION',15X,'VEL',
318      *6X,'VEL',6X,'VEL',1X,6X,'(M)',6X,'(M)',6X,'(M)',13X,'(SIGMA)',3X,
319      *'(M/S)',4X,'(M/S)',4X,'(M/S)',1)
320 C
321 C MAIN COMPUTATIONAL SCHEME
322 C
323      DO 2000 J=1,ITERB
324      DO 2001 I=1,NPTS
325      IF (DP(I).GE. Z)GO TO 24

```

PROGRAM UOUTPLM

```

326 2001 CONTINUE
327 24 LL=I
328 LU=I-1
329 PRODIS=(Z-DP(LU))/(DP(LL)-DP(LU))
330 IF(IFLAG .EQ. 0)GO TO 26
331 TT=TA(LU)+PRODIS*(TA(LL)-TA(LU))
332 SS=SA(LU)+PRODIS*(SA(LL)-SA(LU))
333 DENA=1000.+SIGMAT(SS,TT)
334 GO TO 27
335 26 DENA=DENP(LU)+PRODIS*(DENP(LL)-DENP(LU))
336 27 EINS=E*DENA*DT*B*H*(ABS(UW*TWO*V)/VEL+UW*PI/DD*(U*DB/VEL+B/TWO*
337 *(U/VEL-UO/V1)))
338 ZWEI=TWO*PI*DENA*H*B*ABS(VEL-UW*U/VEL)*A*DT
339 DM=EINS
340 IF (ZWEI .GT. EINS)DM=ZWEI
341 GAMMA=DM/.0069555/PM
342 DT=DT/GAMMA
343 EINS=EINS/GAMMA
344 DM=DM/GAMMA
345 IF(J .GT. 1)GO TO 29
346 IF(I00 .EQ. 2)GO TO 29
347 IF(I00 .EQ. 1 .AND. KFLAG*IFLAG .EQ. 1)WRITE(10,230)X,Z,B,H,
348 *PM,EINS,ZWEI,DIL,DENDIF,U,V,VEL,DEL
349 230 FORMAT(1X,14(1PE9.2))
350 IF(I00 .EQ. 1 .AND. KFLAG*IFLAG .EQ. 0)WRITE(10,230)X,Z,B,H,
351 *PM,EINS,ZWEI,DIL,DENDIF,U,V,VEL
352 IF(I00 .EQ. 0)WRITE(10,231)X,Z,B,DIL,DENDIF,U,V,VEL
353 231 FORMAT(1X,2F9.2,F9.2,F9.1,4F9.2)
354 29 SUM=PM+DM
355 UO=U
356 U=(PM*U+DM*UW)/SUM
357 IF(KFLAG*IFLAG .EQ. 0)GO TO 31
358 T=(PM*T+DM*TT)/SUM
359 S=(PM*S+DM*SS)/SUM
360 DEL=T-TT
361 DEN=1000.+SIGMAT(S,T)
362 GO TO 32
363 31 DEN=(PM*DEN+DM*DENA)/SUM
364 32 DENDIF=DENA-DEN
365 V=PM*V/SUM+DENDIF*G/DEN*DT
366 V1=VEL
367 PM=SUM
368 VEL=SQRT(U*U+V*V)
369 DZ=V*DT
370 DX=U*DT
371 IF(DZ .GT. ZERO)GO TO 35
372 IF(Z .GE. PDEP)GO TO 35
373 ICEASE=2
374 GO TO 90
375 35 DD=VEL*DT
376 BSAVE=B
377 B=SQRT(PM/(DEN*PI*H))
378 DB=B-BSAVE
379 H=VEL/V1*H
380 X=X+DX
381 Z=Z-DZ
382 IF(Z .GT. ZERO)GO TO 37
383 ICEASE=3
384 GO TO 249
385 37 DIL=(RHO*PM)/(PMO*DEN)
386 IF(DENDIF .GT. ZERO)GO TO 80
387 IF(ITRAP .EQ. 1)GO TO 80
388 ITRAP=1
389 JTL=J
390 TTRAP=DENOLD/(DENOLD-DENDIF)

```

PROGRAM UOUTPLM

```

391      ZTRAP=ZOLD+TTRAP*(Z-ZOLD)
392      DILTAP=DILOLD+TTRAP*(DIL-DILOLD)
393      IF(I00 .NE. 2)WRITE(IO,84)
394      84 FORMAT(1X,' *****NORMAL TRAPPING LEVEL REACHED')
395      80 CONTINUE
396      BSOLD=BSNEW
397      BSNEW=B-SPAC2
398      IF(BSNEW .LT. 0)GO TO 82
399      IF(ILAP .EQ. 1)GO TO 82
400      ILAP=1
401      JPM=J
402      IF(ITRAP .EQ. 0)ILAPP=1
403      IF(ITRAP .EQ. 1)ILAPP=2
404      X1=B-BSAVE
405      X2=SPAC2-BSAVE
406      Y1=ZOLD-Z
407      Y2=X2*Y1/X1
408      ZMR=ZOLD-Y2
409      DILM=DILOLD+(DIL-DILOLD)*X2/X1
410      IF(I00 .EQ. 2)GO TO 82
411      IF(ILAPP .EQ. 1)WRITE(IO,240)ZMR,DILM
412      240 FORMAT(1X,' *****PLUMES MERGE AT',F7.2,' M BELOW THE
413      * SURFACE WITH AN AVE. DILUTION OF',F6.1/
414      *1X,' *****FOLLOWING CALCULATIONS DO NOT ACCOUNT FOR MERGING')
415      IF(ILAPP .EQ. 2)WRITE(IO,242)
416      242 FORMAT(1X,' *****PLUMES MERGE, WHICH IS NOT ACCOUNTED FOR
417      * IN THE FOLLOWING CALCULATIONS')
418      IF(JPM .EQ. JTL .AND. JTL .GT. 0)WRITE(IO,241)
419      241 FORMAT(1X,' *****PLUMES MERGED AND TRAPPED AT THE SAME TIME')
420      82 CONTINUE
421      DZZ=DZOLD
422      DZOLD=DZ
423      ZOLDZ=ZOLD
424      ZOLD=Z
425      DENOT=DENOLD
426      DENOLD=DENDIF
427      DILZ=DILOLD
428      DILOLD=DIL
429      IF(J .EQ. 1)GO TO 81
430      IF(J/IR -(J-1)/IR .NE. 1)GO TO 2000
431      81 IF(KFLAG*IFLAG .EQ. 1 .AND. I00 .EQ. 1)WRITE(IO,230)X,Z,B,H,
432      *PM,EINS,ZWEI,DIL,DENDIF,U,V,VEL,DEL
433      IF(KFLAG*IFLAG .EQ. 0 .AND. I00 .EQ. 1)WRITE(IO,230)X,Z,B,H,
434      *PM,EINS,ZWEI,DIL,DENDIF,U,V,VEL
435      IF(I00 .EQ. 0)WRITE(IO,231)X,Z,B,DIL,DENDIF,U,V,VEL
436      2000 CONTINUE
437      ICEASE=4
438      90 CONTINUE
439      C
440      C:::::PRINT FINAL RESULTS
441      C
442      IF(KFLAG*IFLAG .EQ. 1 .AND. I00 .EQ. 1)WRITE(IO,230)X,Z,B,H,
443      *PM,EINS,ZWEI,DIL,DENDIF,U,V,VEL,DEL
444      IF(KFLAG*IFLAG .EQ. 0 .AND. I00 .EQ. 1)WRITE(IO,230)X,Z,B,H,
445      *PM,EINS,ZWEI,DIL,DENDIF,U,V,VEL
446      IF(I00 .EQ. 0)WRITE(IO,231)X,Z,B,DIL,DENDIF,U,V,VEL
447      WRITE(IO,250)J
448      IF(ICEASE .EQ. 4)WRITE(IO,254)ITERB
449      IF(ICEASE .EQ. 4)GO TO 91
450      IF(ICEASE .EQ. 2)WRITE(IO,252)
451      C
452      C:::::IF THE VERTICAL VELOCITY WENT THRU ZERO, BUT DID NOT TRAP,
453      C      CALCULATE THE DILUTION WHEN VERTICAL VELOCITY WAS ZERO.
454      C
455      IF(ITRAP .EQ. 0 .AND. ICEASE .EQ. 2)

```

PROGRAM UOUTPLM

```

456      *DILDZ=DILOLD+DZOLD*((DILOLD-DILZ)/(DZZ-DZOLD))
457      249 CONTINUE
458      IF(ICEASE .EQ. 3)WRITE(IO,253)
459      C
460      C:::::IF THE PLUME SURFACED BUT DID NOT TRAP, CALCULATE THE
461      C      DILUTION AT THE SURFACE.
462      C
463      IF(ITRAP .EQ. 0 .AND. ICEASE .EQ. 3)
464      *DILSUR=DILOLD+ZOLD*((DILOLD-DILZ)/(ZOLDZ-ZOLD))
465      C
466      C:::::IF JPM=JTL AND NOT ZERO, MERGING AND TRAPPING
467      C      OCCURRED AT THE SAME TIME.
468      C
469      IF(JPM .EQ. JTL .AND. JTL .GT. 0)GO TO 264
470      C
471      IF(ITRAP .EQ. 0 .AND. ILAP .EQ. 0)WRITE(IO,248)
472      IF(ITRAP .EQ. 0 .AND. ILAP .EQ. 1 .AND. ICEASE .EQ. 2)
473      *WRITE(IO,246)
474      IF(ITRAP .EQ. 0 .AND. ILAP .EQ. 1 .AND. ICEASE .EQ. 3)
475      *WRITE(IO,263)
476      IF(ITRAP .EQ. 1 .AND. ILAPP .EQ. 1)WRITE(IO,261)
477      IF(ITRAP .EQ. 1 .AND. ILAPP .EQ. 2)WRITE(IO,267)
478      IF(JPM .LT. JTL .AND. IOO .EQ. 2)WRITE(IO,268)DILM,ZMR
479      IF(ITRAP .EQ. 1)WRITE(IO,260)ZTRAP,DILTAP
480      IF(ITRAP .EQ. 0 .AND. ICEASE .EQ. 2)WRITE(IO,247)DILDZ
481      IF(ITRAP .EQ. 0 .AND. ICEASE .EQ. 3)WRITE(IO,262)DILSUR
482      GO TO 91
483      C
484      264 CONTINUE
485      WRITE(IO,265)ZTRAP,DILTAP
486      C
487      91 CONTINUE
488      IF(INTER .EQ. 0)GO TO 70
489      IF(ICEASE .EQ. 4)WRITE(IT,254)ITERB
490      IF(ICEASE .EQ. 4)GO TO 70
491      IF(ICEASE .EQ. 2)WRITE(IT,252)
492      IF(ICEASE .EQ. 3)WRITE(IT,253)
493      IF(JPM .EQ. JTL .AND. JTL .GT. 0)GO TO 266
494      IF(ITRAP .EQ. 0 .AND. ILAP .EQ. 0)WRITE(IT,248)
495      IF(ITRAP .EQ. 0 .AND. ILAP .EQ. 1 .AND. ICEASE .EQ. 2)
496      *WRITE(IT,246)
497      IF(ITRAP .EQ. 0 .AND. ILAP .EQ. 1 .AND. ICEASE .EQ. 3)
498      *WRITE(IT,263)
499      IF(ITRAP .EQ. 1 .AND. ILAPP .EQ. 1)WRITE(IT,261)
500      IF(ITRAP .EQ. 1 .AND. ILAPP .EQ. 2)WRITE(IT,267)
501      IF(JPM .LT. JTL .AND. IOO .EQ. 2)WRITE(IT,268)DILM,ZMR
502      IF(ITRAP .EQ. 1)WRITE(IT,260)ZTRAP,DILTAP
503      IF(ITRAP .EQ. 0 .AND. ICEASE .EQ. 2)WRITE(IT,247)DILDZ
504      IF(ITRAP .EQ. 0 .AND. ICEASE .EQ. 3)WRITE(IT,262)DILSUR
505      GO TO 70
506      266 WRITE(IT,265)ZTRAP,DILTAP
507      C
508      C
509      246 FORMAT(/1X,' PLUMES MERGED BEFORE VERTICAL VELOCITY WENT
510      * THRU ZERO')
511      247 FORMAT(1X,' WHEN V=0, DILUTION =',F10.2)
512      248 FORMAT(/1X,' PLUMES DID NOT TRAP OR MERGE')
513      250 FORMAT(/1X,' NUMBER OF STEPS='15)
514      252 FORMAT(/1X,' COMPUTATIONS CEASE: VERTICAL PLUME VELOCITY
515      * WENT THRU ZERO')
516      253 FORMAT(/1X,' COMPUTATIONS CEASE: PLUME HITS WATER SURFACE')
517      254 FORMAT(/1X,' COMPUTATIONS CEASE: NUMBER OF STEPS EXCEEDS',15,
518      * (PARAMETER ITERB)')
519      260 FORMAT(/1X,' TRAPPING LEVEL=',F7.2,' M BELOW WATER SURFACE,
520      * DILUTION=',F7.2)

```

PROGRAM UOUTPLM

```

521 261 FORMAT(1X,' PLUMES MERGED BEFORE TRAPPING LEVEL REACHED')
522 262 FORMAT(1X,' PLUME SURFACES, DILUTION =',F10.2)
523 263 FORMAT(1X,' PLUMES MERGED BEFORE REACHING SURFACE')
524 265 FORMAT(1X,' PLUMES MERGED AND TRAPPED AT THE SAME TIME.'/
525 *1X,' TRAPPING LEVEL =',F7.2,' M BELOW SURFACE; DILUTION =',F7.2)
526 267 FORMAT(1X,' PLUMES MERGED AFTER TRAPPING LEVEL REACHED')
527 268 FORMAT(1X,' NOTE: AVERAGE DILUTION WAS',F6.1,' WHEN PLUMES'/
528 *1X,' MERGED AT',F7.2,' M BELOW THE WATER SURFACE'/
529 *1X,' TRAPPING LEVEL NOT YET REACHED. DILUTION'/
530 *1X,' SHOWN BELOW DOES NOT ACCOUNT FOR MERGING.')
```

70 CONTINUE

```

532 IF(INTER .EQ. 1)GO TO 3
533 GO TO 85
534 END
```

C

```

535 SUBROUTINE ROUTP(K,LIMIT)
537 C
538 C READ UNIVERSAL DATA FILE AND CONVERT DATA FOR USE IN PROGRAM UOUTPLM
539 C
```

```

540 COMMON/WRITE/DPW(30),SAW(30),TAW(30),DENPW(30),DENPPW(30)
541 COMMON/AVAR/DP(30),SA(30),TA(30),DENP(30),DENPP(30)
542 COMMON/VAR/INTER,IDFP,ICUTOP,IOI,I00,IHF,IEF,IAF,IIF,IIRF,QT,NP,
543 *PDIA,VANG,PDEP,UW,IR,HANG,SPACE,H,E,A,ITERB,NPTS,T,S,DEN,FR
544 COMMON/SAV/II(5)
545 COMMON/COUNT/NMO,NW
546 COMMON/INOUT/IN,IO,IT
547 REAL*8 TITLE(10),TITLI(10),AA(10)
548 BYTE IFILE(20),OFILE(20)
549 COMMON/TVAR/TITLE,TITLI,IFILE
550 DATA IFILE(20)/O/,OFILE(20)/O/
551 IF(K .EQ. 2)GO TO 40
552 READ(IN,200,END=99,ERR=999)(TITLE(I),I=1,10)
553 200 FORMAT(10A8)
554 READ(IN,201,ERR=999)INTER,IDFP,ICUTOP,(II(J),J=1,5)
555 201 FORMAT(8I2)
556 READ(IN,202,ERR=999)QT,NP,PDIA,VANG,PDEP
557 202 FORMAT(F10.0,I10,3F10.0)
558 READ(IN,203,ERR=999)UW,HANG,SPACE
559 203 FORMAT(3F10.0)
```

C

```

561 C ALL *X* VARIABLES ARE IGNORED
562 C
```

```

563 IF(ICUTOP .EQ. 1)READ(IN,213,ERR=999)AX,IX1,IX2,IX3,IX4,
564 *IX5,PX,RX,DX,H,E,A,ITERB,IR
565 213 FORMAT(F5.0,2I5,3I2,6F5.0,2I5)
566 C
567 C
```

```

568 READ(IN,204,ERR=999)NPTS,S,T
569 204 FORMAT(I10,2F10.0)
570 READ(IN,205,ERR=999)((DP(I),SA(I),TA(I)),I=1,NPTS)
571 205 FORMAT(3F10.0)
572 IF(ICUTOP .EQ. 0)GO TO 30
573 IHF=1
574 IEF=1
575 IAF=1
576 IIF=1
577 IIRF=1
578 IF(H)21,21,22
579 21 H=PDIA*.5
580 IHF=0
581 22 IF(E)23,23,24
582 23 E=1.0
583 IEF=0
584 24 IF(A)25,25,26
585 25 A=0.1
```

PROGRAM UOUTPLM

```

586      IAF=0
587      26 IF(ITERB)27,27,28
588      27 ITERB=5000
589      IIF=0
590      28 IF(IR)29,29,31
591      29 IR=50
592      IIRF=0
593      GO TO 31
594      30 IHF=0
595      IEF=0
596      IAF=0
597      IIF=0
598      IIRF=0
599      H=PDIA*0.5
600      E=1.0
601      A=0.1
602      ITERB=5000
603      IR=50
604      31 IOI=II(2)
605      IOO=IOI
606      50 CONTINUE
607      RETURN
608      40 NW=NPTS+5+ICUTOP
609      IF(LIMIT .EQ. 0 .AND. IDFP .EQ. 1)WRITE(IO,403)
610      403 FORMAT(1H1)
611      IF(IDFP .EQ. 0)GO TO 50
612      WRITE(IO,400)(IFILE(IK),IK=1,19)
613      400 FORMAT(/1X,' UNIVERSAL DATA FILE: ',19A1/)
614      IF(INTER .EQ. 1)WRITE(IO,10)
615      10 FORMAT(
616      *1X,' *****'/
617      *1X,' * NOTE, THIS IS THE ORIGINAL FILE. */
618      *1X,' * IT DOES NOT REFLECT CHANGES MADE INTERACTIVELY. */
619      *1X,' * THOSE CHANGES ARE SHOWN IN THE OUTPUT HEADING. */
620      *1X,' *****'/)
621      REWIND IN
622      IF(NWO .EQ. 0)GO TO 41
623      DO 1001 I=1,NWO
624      READ(IN,401)(AA(J),J=1,10)
625      1001 CONTINUE
626      41 DO 1000 I=1,NW
627      READ(IN,401)(AA(J),J=1,10)
628      401 FORMAT(10A8)
629      WRITE(IO,402)(AA(J),J=1,10)
630      402 FORMAT(1X,10A8)
631      1000 CONTINUE
632      GO TO 50
633      999 WRITE(IO,403)
634      WRITE(IO,998)(TITLE(IK),IK=1,10)
635      998 FORMAT(1X,' COMPUTATIONS CEASE FOR: '/2X,'CASE I.D. ',10A8/
636      *1X,' INPUT CONVERSION ERROR, CHECK DATA FILE')
637      IF(INTER .EQ. 1)WRITE(IT,998)(TITLE(IK),IK=1,10)
638      99 CALL EXIT
639      END
640      C
641      SUBROUTINE NOUTP(K,IFLAG)
642      C
643      C PRINTS INPUT DATA FOR UOUTPLM IN NW,NW FORMAT
644      C
645      COMMON/WRITE/DPW(30),SAW(30),TAW(30),DENPW(30),DENPPW(30)
646      COMMON/AVAR/DP(30),SA(30),TA(30),DENP(30),DENPP(30)
647      COMMON/VAR/INTER,IDFP,ICUTOP,IOI,IOO,IHF,IEF,IAF,IIF,IIRF,QT,NP,
648      *PDIA,VANG,PDEP,UW,IR,HANG,SPACE,H,E,A,ITERB,NPTS,T,S,DEN,FR
649      REAL*8 TITLE(10),TITLI(10)
650      BYTE IFILE(20),OFILE(20)

```

PROGRAM UOUTPLM

```

651      COMMON/TVAR/TITLE,TITLI,IFILE
652      COMMON/INOUT/IN,IO,IT
653      DATA IFILE(20)/0/,OFILE(20)/0/
654  C
655      IF(K-2)1,2,3
656  1      IF(INF .EQ. 0)WRITE(IO,111)
657          IF(INF .EQ. 1)WRITE(IO,112)H
658          IF(IEF .EQ. 0)WRITE(IO,113)E
659          IF(IEF .EQ. 1)WRITE(IO,114)E
660          IF(IAF .EQ. 0)WRITE(IO,115)A
661          IF(IAF .EQ. 1)WRITE(IO,116)A
662          IF(IIF .EQ. 0)WRITE(IO,117)ITERB
663          IF(IIF .EQ. 1)WRITE(IO,118)ITERB
664          IF(IOO .EQ. 2)GO TO 128
665          IF(IIRF .EQ. 0)WRITE(IO,119)IR
666          IF(IIRF .EQ. 1)WRITE(IO,120)IR
667  128  WRITE(IO,122)UW
668          IF(T .NE. 0)WRITE(IO,123)T
669          IF(T .NE. 0.)WRITE(IO,124)S
670          DEN1=DEN-1000.
671          WRITE(IO,125)DEN1
672          WRITE(IO,126)FR
673          IF(IFLAG .EQ. 0)GO TO 134
674          WRITE(IO,131)
675          WRITE(IO,132)((DPW(I),SAW(I),TAN(I),DENPPW(I)),I=1,NPTS)
676          GO TO 2
677  134  WRITE(IO,135)
678          WRITE(IO,133)((DPW(I),DENPPW(I)),I=1,NPTS)
679  2      WRITE(IO,200)
680          WRITE(IO,201)QT
681          WRITE(IO,202)NP
682          WRITE(IO,203)PDIA
683          IF(NP .GT. 1)WRITE(IO,121)SPACE
684          IF(NP .EQ. 1)WRITE(IO,127)
685          WRITE(IO,204)VANG
686          WRITE(IO,205)PDEP
687          RETURN
688  3      CONTINUE
689          WRITE(IO,210)(TITLI(IK),IK=1,10)
690          WRITE(IO,126)FR
691          GO TO 2
692  C
693  111  FORMAT(/2X,'INITIAL THICKNESS OF PLUME ELEMENT = PORT RADIUS '
694      *,'(DEFAULT)')
695  112  FORMAT(/2X,'INITIAL THICKNESS OF PLUME ELEMENT =' ,F9.2, ' M')
696  113  FORMAT(2X,'IMPINGEMENT ENTRAINMENT COEFFICIENT =' ,F9.2,4X,
697      *,'(DEFAULT)')
698  114  FORMAT(2X,'IMPINGEMENT ENTRAINMENT COEFFICIENT =' ,F9.2)
699  115  FORMAT(2X,'ASPIRATION ENTRAINMENT COEFFICIENT =' ,F9.2,4X,
700      *,'(DEFAULT)')
701  116  FORMAT(2X,'ASPIRATION ENTRAINMENT COEFFICIENT =' ,F9.2)
702  117  FORMAT(2X,'NUMBER OF STEPS ALLOWED',13X,'=' ,I6,7X,'(DEFAULT)')
703  118  FORMAT(2X,'NUMBER OF STEPS ALLOWED',13X,'=' ,I6)
704  119  FORMAT(2X,'PRINTOUT INTERVAL',19X,'=' ,I6,7X,'(DEFAULT)')
705  120  FORMAT(2X,'PRINTOUT INTERVAL',19X,'=' ,I6)
706  121  FORMAT(2X,'PORT SPACING',24X,'=' ,F9.2, ' M')
707  122  FORMAT(/2X,'AMBIENT CURRENT SPEED',15X,'=' ,F9.2, ' M/S')
708  123  FORMAT(2X,'INITIAL TEMPERATURE OF THE PLUME',4X,'=' ,F9.2,4X,
709      *,'DEGREES CENTIGRADE')
710  124  FORMAT(2X,'INITIAL SALINITY OF THE PLUME',7X,'=' ,F9.2, ' PPT')
711  125  FORMAT(2X,'INITIAL DENSITY OF THE PLUME',8X,'=' ,F11.4,
712      *, ' SIGMAT UNITS')
713  126  FORMAT(1X,' FROUDE NUMBER',23X,'=' ,F8.1)
714  127  FORMAT(2X,'PORT SPACING',24X,'=' 1000.0 ' M(DEFAULT)')
715  131  FORMAT(/3X,'DEPTH',4X,'SALIN',5X,'TEMP',3X,'SIGMAT',/,4X,

```

PROGRAM UOUTPLM

```

716      *'(M)',5X,'(PPT)',5X,'(C)',/)
717      132 FORMAT(1X,F7.2,F9.2)
718      133 FORMAT(1X,F7.2,F9.2)
719      135 FORMAT(/3X,'DEPTH',3X,'SIGMAT'/4X,'(M)')
720      200 FORMAT(1X)
721      201 FORMAT(2X,'TOTAL EFFLUENT FLOW',17X,'=',F11.4,' CMS')
722      202 FORMAT(2X,'NUMBER OF PORTS',21X,'=',I6)
723      203 FORMAT(2X,'PORT DIAMETER',23X,'=',F11.4,' M')
724      204 FORMAT(2X,'VERTICAL PORT ANGLE FROM HORIZONTAL =',F8.1,5X,
725      *'DEGREES')
726      205 FORMAT(2X,'PORT DEPTH',26X,'=',F9.2,' M')
727      210 FORMAT(/2X,70('X'),/2X,'RUN TITLE: ',10A8)
728      END
729      C
730      SUBROUTINE LIMITS(KK,PDEP,VANG,T,S,NPTS,INTER,TITLE,TITLI,
731      *IEXIT,LIMIT)
732      C
733      C      THIS SUBROUTINE CHECKS LIMITS OF PORT DEPTH (GREATER THAN 0. AND
734      C      LESS THAN OR EQUAL TO PROFILE DEPTH), DISCHARGE ANGLE (EQUAL TO
735      C      OR GREATER THAN -5 DEG BUT LESS THAN OR EQUAL TO 90 DEG,
736      C      VERTICAL, AND IF NEGATIVE, THAT THERE IS AT LEAST ONE PROFILE
737      C      DATA POINT AT LEAST ONE METER BELOW THE PORT DEPTH). THAT THE
738      C      EFFLUENT DENSITY IS LESS THAN THE AMBIENT DENSITY AND THAT THERE
739      C      ARE AMBIENT PROFILE VALUES FOR THE SURFACE, IF INTER=1, THE PORT
740      C      DEPTH, ANGLE AND EFFLUENT DENSITY CAN BE CORRECTED INTERACTIVELY
741      C      BUT SURFACE DATA AND PROFILE DEPTH CORRECTIONS MUST BE MADE TO
742      C      THE DATA SET(S) AND REENTERED. IF INTER=0, ALL CORRECTION MUST BE
743      C      MADE TO THE DATA SET(S) AND REENTERED.
744      C
745      COMMON/WRITE/DPW(30),SAW(30),TAW(30),DENPW(30),DENPPW(30)
746      COMMON/INOUT/IN,IO,IT
747      REAL*8 TITLE(10),TITLI(10)
748      LIMIT=0
749      IDEEP=0
750      IVANG=0
751      NEGANG=0
752      IRHO=0
753      INO=0
754      IEXIT=0
755      100 IF(PDEP .EQ. 0. .OR. DPW(NPTS) .LT. PDEP)GO TO 1
756      10 IF(VANG .LT. 0. .OR. VANG .GT. 90.)GO TO 101
757      30 IF(T .EQ. 0.)RHOE=S
758      IF(T .NE. 0.)RHOE=1.+0.001*SIGMAT(S,T)
759      DO 31 J=2,NPTS
760      IF(PDEP .LE. DPW(J))GO TO 32
761      31 CONTINUE
762      GO TO 40
763      32 JK=J
764      PRO=(PDEP-DPW(JK-1))/(DPW(JK)-DPW(JK-1))
765      TT=TAW(JK-1)+PRO*(TAW(JK)-TAW(JK-1))
766      SS=SAW(JK-1)+PRO*(SAW(JK)-SAW(JK-1))
767      IF(TAW(JK) .EQ. 0.)RHOA=SS
768      IF(TAW(JK) .NE. 0.)RHOA=1.+0.001*SIGMAT(SS,TT)
769      IF(RHOA .GT. RHOE)GO TO 40
770      IRHO=1
771      LIMIT=1
772      IF(INTER .EQ. 0)GO TO 40
773      120 CONTINUE
774      WRITE(IT,33)RHOE,RHOA
775      33 FORMAT(/1X,' EFFLUENT DENSITY ('F7.5,' G/CM3) MUST BE'/
776      *1X,' LESS THAN THE AMBIENT DENSITY ('F7.5,' G/CM3)')
777      WRITE(IT,4)
778      READ(IT,5)NANS
779      IF(NANS .EQ. 'N')GO TO 40
780      IF(NANS .NE. 'Y')WRITE(IT,200)

```

PROGRAM UOUTPLM

```

781      IF(NANS .NE. 'Y')GO TO 120
782      IRHO=2
783      IF(T .EQ. 0.)GO TO 36
784      WRITE(IT,34)
785      34 FORMAT(/1X,' NEW EFFLUENT TEMP VALUE? ',S)
786      READ(IT,7)T
787      WRITE(IT,35)
788      35 FORMAT(1X,' NEW EFFLUENT SAL VALUE? ',S)
789      READ(IT,7)S
790      GO TO 30
791      36 WRITE(IT,37)
792      37 FORMAT(/1X,' NEW EFFLUENT DENSITY VALUE? ',S)
793      READ(IT,7)S
794      GO TO 30
795      101 IF(VANG .LT. -5. .OR. VANG .GT. 90.)GO TO 11
796      C
797      C:::::IF CONTROL IS NOT PASSED TO 11, VANG .GE. -5. BUT .LT. 0.
798      C      BECAUSE IT'S NEGATIVE, PROFILE DEPTH MUST BE AT LEAST ONE
799      C      METER DEEPER THAN THE PORT DEPTH.
800      C
801      IF(DPW(NPTS)-PDEP .GE. 1.0)GO TO 30
802      NEGANG=1
803      C
804      1 IDEEP=1
805      LIMIT=1
806      IF(INTER .EQ. 0)GO TO 10
807      121 CONTINUE
808      IF(PDEP .EQ. 0.)WRITE(IT,2)
809      2 FORMAT(/1X,' ZERO DISCHARGE DEPTH NOT ALLOWED,
810      * CHANGE PORT DEPTH.')
811      IF(PDEP .GT. 0. .AND. DPW(NPTS) .LT. PDEP)
812      *WRITE(IT,3)PDEP,DPW(NPTS)
813      3 FORMAT(/1X,' PORT DEPTH ('F6.2,' M) IS DEEPER THAN'/
814      *1X,' THE PROFILE DEPTH ('F6.2,' M), CHANGE PORT DEPTH.')
815      IF(NEGANG .EQ. 1)WRITE(IT,21)PDEP,DPW(NPTS)
816      21 FORMAT(/1X,' FOR A NEGATIVE DISCHARGE ANGLE, THE PORT DEPTH ('
817      *F3.4,' M) MUST BE AT'/1X,' LEAST ONE METER LESS THAN THE PROFILE
818      * DEPTH ('F3.4,' M), CHANGE PORT DEPTH.')
819      WRITE(IT,4)
820      4 FORMAT(/1X,' WANT TO MAKE CORRECTION? YES/NO ',S)
821      READ(IT,5)NANS
822      5 FORMAT(A1)
823      IF(NANS .EQ. 'N')GO TO 10
824      IF(NANS .NE. 'Y')WRITE(IT,200)
825      IF(NANS .NE. 'Y')GO TO 121
826      IDEEP=2
827      WRITE(IT,6)
828      6 FORMAT(1X,' NEW VALUE? ',S)
829      READ(IT,7)PDEP
830      7 FORMAT(F7.0)
831      GO TO 100
832      11 IVANG=1
833      LIMIT=1
834      IF(INTER .EQ. 0)GO TO 30
835      122 CONTINUE
836      WRITE(IT,12)VANG
837      12 FORMAT(/1X,' DISCHARGE ANGLE ('F6.1,' ) IS NOT IN THE
838      * RANGE OF -5 TO 90 DEG')
839      WRITE(IT,4)
840      READ(IT,5)NANS
841      IF(NANS .EQ. 'N')GO TO 30
842      IF(NANS .NE. 'Y')WRITE(IT,200)
843      IF(NANS .NE. 'Y')GO TO 122
844      IVANG=2
845      WRITE(IT,6)

```

PROGRAM UOUTPLM

```

845      READ(IT,7)VANE
846      GO TO 10
847
848      40 IF(BPW(1) .EQ. 0.)GO TO 50
849      INQ=1
850      LIMIT=1
851      IF(INTER .EQ. 0)GO TO 50
852      WRITE(IT,41)(TITLE(K),K=1,10)
853      41 FORMAT(1H1/1X,' COMPUTATIONS CEASE FOR'/1X,' CASE I.D. ',10A8)
854      WRITE(IT,44)(TITLI(K),K=1,10)
855      44 FORMAT(1X,' RUN TITLE: ',10A8)
856      WRITE(IT,42)
857      42 FORMAT(/1X,' NO SURFACE DATA FOR THE AMBIENT DENSITY/CURRENT
858      * PROFILE.'/1X,' WITHOUT THESE DATA, COMPUTATIONAL ERRORS
859      * MAY OCCUR.'/1X,' MAKE CORRECTIONS TO DATA SET(S) AND REENTER.')
```

```

860      WRITE(IT,43)
861      43 FORMAT(/1X,' GOING TO NEXT DATA SET IF THERE IS ONE.')
```

```

862      50 IF(LIMIT .EQ. 0)GO TO 70
863      IF(INTER .EQ. 1)GO TO 69
864      51 WRITE(IO,41)(TITLE(K),K=1,10)
865      IF(INTER .EQ. 0)GO TO 73
866      IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
867      WRITE(IT,41)(TITLE(K),K=1,10)
868      IF(KK .EQ. 2)WRITE(IT,44)(TITLI(K),K=1,10)
869      WRITE(IT,74)
870      74 FORMAT(/1X,' NECESSARY CORRECTIONS NOT MADE!')
```

```

871      WRITE(IT,43)
872      73 WRITE(IO,67)
873      67 FORMAT(/1X,' CORRECT THE FOLLOWING AND REENTER DATA.')
```

```

874      IEXIT=1
875      68 IF(IDEEP .EQ. 0)GO TO 52
876      IF(IDEEP .EQ. 1)WRITE(IO,56)PDEP
877      IF(IDEEP .EQ. 2)WRITE(IO,57)PDEP
878      52 IF(IVANG .EQ. 0)GO TO 54
879      IF(IVANG .EQ. 1)WRITE(IO,58)VANG
880      IF(IVANG .EQ. 2)WRITE(IO,59)VANG
881      54 IF(IRHO .EQ. 0)GO TO 55
882      IF(IRHO .EQ. 1)WRITE(IO,64)
883      IF(IRHO .EQ. 2)WRITE(IO,65)RHOE
884      55 IF(INQ .EQ. 0)GO TO 75
885      WRITE(IO,42)
886      WRITE(IO,43)
887      GO TO 70
888      69 IF(IDEEP .EQ. 1 .OR. IVANG .EQ. 1 .OR. IRHO .EQ. 1
889      *.OR. INQ .EQ. 1)GO TO 51
890      WRITE(IO,66)(TITLE(K),K=1,10)
891      IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
892      WRITE(IO,71)
893      GO TO 68
894      56 FORMAT(
895      *1X,' PORT DEPTH WAS ENTERED AS ',F6.2,' METERS:/'
896      *1X,' IF THE DISCHARGE ANGLE IS POSITIVE, THE PORT DEPTH MUST/'
897      *1X,' BE GREATER THAN ZERO BUT LESS THAN OR EQUAL TO THE PROFILE
898      * DEPTH./'
899      *1X,' IF THE DISCHARGE ANGLE IS NEGATIVE, THE PORT DEPTH MUST/'
900      *1X,' BE AT LEAST ONE METER LESS THAN THE PROFILE DEPTH.')
```

```

901      57 FORMAT(1X,' PORT DEPTH CHANGED TO: ',F6.2,' M')
```

```

902      58 FORMAT(1X,' DISCHARGE ANGLE (',F6.1,')
903      * MUST BE .GE. -5 BUT .LE. 90 DEG')
```

```

904      59 FORMAT(1X,' DISCHARGE ANGLE CHANGED TO: ',F4.1,' DEG')
```

```

905      64 FORMAT(1X,' EFFLUENT DENSITY MUST BE .LT. AMBIENT
906      * DENSITY AT THE DISCHARGE DEPTH')
```

```

907      65 FORMAT(1X,' EFFLUENT DENSITY CHANGED TO: ',F7.5,' G/CM3')
```

```

908      66 FORMAT(1H1/1X,' COMPUTATIONS CONTINUE FOR'/1X,' CASE I.D. ',10A8)
909      71 FORMAT(1X,' CORRECTIONS WERE INTERACTIVELY MADE TO
910      * THE FOLLOWING:')
```

PROGRAM UOUTPLN

```

911 200 FORMAT(/1X,' *****YOU MADE A MISTAKE, TRY AGAIN*****')
912 75 IF(IEXIT .EQ. 1)WRITE(10,43)
913 70 RETURN
914 END
915 C
916 SUBROUTINE USORT(DP,X,Y,Z,NPTS)
917 DIMENSION DP(30),X(30),Y(30),Z(30)
918 NESTED=NPTS
919 L=NESTED-1
920 DO 1000 M=1,L
921     NESTED=NESTED-1
922     DO 1000 I=1,NESTED
923         IF(DP(I) .LE. DP(I+1))GO TO 1000
924         DUMMY=DP(I)
925         DP(I)=DP(I+1)
926         DP(I+1)=DUMMY
927         DUMMY=X(I)
928         X(I)=X(I+1)
929         X(I+1)=DUMMY
930         DUMMY=Y(I)
931         Y(I)=Y(I+1)
932         Y(I+1)=DUMMY
933         DUMMY=Z(I)
934         Z(I)=Z(I+1)
935         Z(I+1)=DUMMY
936 1000 CONTINUE
937 RETURN
938 END
939 C
940 FUNCTION SIGMAT(SAL,T)
941 SIGD=((6.8E-6*SAL)-4.82E-4)*SAL+.8149)*SAL-0.093
942 B=1.E-6*T*((.01667*T-.8164)*T+18.03)
943 A=.001*T*((.0010843*T-.09818)*T+4.7867)
944 SURT=(T-3.98)*(T-3.98)*(T+283.)/(503.57*(T+67.26))
945 SIGMAT=(SIGD+.1324)*(1.-A+B*(SIGD-.1324))-SURT
946 RETURN
947 END

```

SECTION 4
UDKHDEN Listing

PROGRAM UDKHDE

```

1      PROGRAM UDKHDE
2      C
3      C THIS IS PROGRAM UDKHDE WRITTEN FOR A PDP 11/70 (FORTRAN IV-PLUS).
4      C IN THE ABOVE, PROGRAM NAME MUST NOT BE MORE THAN SIX CHARACTERS.
5      C
6      C UP-TO-DATE AS OF AUGUST 1985 (NO CHANGES SINCE 6-19-85).
7      C JUNE 19, 1985 CHANGE: CO TO COE IN TWO WRITE LISTS, ONE IS THE
8      C OUTPUT HEADING AND THE OTHER IS THE INTERACTIVE LIST. ADDED TWO
9      C IF STATEMENTS SETTING COE=CO AND TOE=TO FOLLOWING INPUT CHANGES
10     C IN THE INTERACTIVE MODE. ADDED THE BOXED NOTE FOLLOWING THE TITLE
11     C OF THE UNIVERSAL DATA FILE NAME. REWROTE THE OUTPUT MESSAGE WHEN
12     C PORT DEPTH WAS TOO DEEP.
13     C APRIL 17, 1985 MODIFIED PROGRAM TO ACCEPT A NEGATIVE DISCHARGE
14     C ANGLE I.E. VANG .GE. -5 .AND. .LE. 130 DEG. THIS REQUIRED CHANGES
15     C IN SUBROUTINES ZFE (ADDED ONE STATEMENT) AND LIMITS (ADDED SEVERAL
16     C STATEMENTS AND CHANGED SOME). ALSO, REMOVED MANY STATEMENTS FROM
17     C SUBROUTINE OUTP1 THAT CALCULATED VARIABLES IN THE ZONE OF FLOW
18     C ESTABLISHMENT AND OUTPUT THEM IN THE OLD VERSION (DKHPLM).
19     C MARCH 19, 1985 ADDED STATEMENTS IN SUBROUTINE LIMITS TO DETECT
20     C ERRONEOUS RESPONSES TO PROGRAM PROMPTS I.E. .NE. TO YES OR NO.
21     C MARCH 12, 1985. ADDED A WRITE *****PROGRAM UDKHDE, ??? 1985*****
22     C FEBRUARY 20, 1985. CHANGED GM/CC TO G/CM3.
23     C JANUARY 28, 1985. CONVERTED FROM SINGLE TO DOUBLE PRECISION
24     C AND CHANGED FOUR OUTPUT "IF(" STATEMENTS INVOLVING ISURF AND ITRAP.
25     C JANUARY 23, 1985. CALL USORT ARGUMENT PI CHANGED TO ZUA.
26     C JANUARY 17, 1985. COMMENTS ADDED RE LOGICAL UNITS "IN", "IOUT"
27     C AND "ITERM". ALL "READ(3," CHANGED TO "READ(IN," AND ALL READ(5
28     C AND WRITE(5 CHANGED TO READ(ITERM AND WRITE(ITERM).
29     C REVISED OCTOBER 10, 1984.
30     C LIMIT OF THE VERTICAL DISCHARGE ANGLE (VANG) CHANGED TO RANGE
31     C FROM GREATER THAN OR EQUAL TO 0 DEG TO LESS THAN OR EQUAL TO
32     C 130 DEG. ALSO CHANGED THE SUBROUTINE SIGMAT (ONE PLACE) FROM
33     C (T-3.98)**2 TO (T-3.98)*(T-3.98). THE LATTER GIVES CLOSER
34     C AGREEMENT WITH H.O. PUB. NO. 615. TABLES FOR SEA WATER DENSITY.
35     C
36     C THIS PROGRAM CALCULATES THE CHARACTERISTICS OF A LINE OF EQUALLY
37     C SPACED BUOYANT DISCHARGES INTO A FLOWING STRATIFIED AMBIENT WATERS.
38     C THE METHOD OF SOL'N INVOLVES 7 ORDINARY DIFFERENTIAL EQUATIONS
39     C WHICH ARE 1. CONSERVATION OF MASS 2. CONSERVATIONS OF ENERGY
40     C 3. CONSERVATION OF CONCENTRATION 4. DENSITY DEFICIENCY
41     C 5.6.7. MOMENTUM EQNS IN THE Z (AXIAL), K (VERTICAL) J (HORIZONTAL
42     C AND PARALLEL TO AMBIENT CURRENT DIRECTION) EQN 4 IS USED ONLY
43     C IN ALGEBRAIC FORM. THE SIX OTHER EQUATIONS ARE WRITTEN FOR
44     C A CONTROL VOLUME WHICH IS FINITE IN A DIRECTION PERPENDICULAR
45     C TO THE JET AND INFINITESIMAL IN THE DIRECTION OF THE JET AXIS.
46     C OUTPUT IN THE ZONE OF FLOW ESTABLISHMENT HAS BEEN SUPPRESSED.
47     C
48     C
49     C
50     C      ***** THREE INPUT OPTIONS ARE AVAILABLE *****
51     C
52     C      1. THE USER MAY ENTER THE TEMPERATURE AND SALINITY OF THE
53     C      DISCHARGE AND AMBIENT. DENSITIES ARE CALCULATED INTERNALLY.
54     C
55     C      2. OR THE USER MAY ENTER THE DENSITY OF THE
56     C      DISCHARGE AND AMBIENT DIRECTLY.
57     C
58     C      3. OR THE USER MAY ENTER THE TEMPERATURE AND SALINITY
59     C      OF EITHER ONE AND THE DENSITY OF THE OTHER.
60     C
61     C THIS IS A UNIVERSAL DATA FILE (UDF). IT IS THE INPUT FILE
62     C PRESENTLY USED BY FOUR OTHER PLUME PROGRAMS.
63     C INPUT CONSISTS OF FIVE (OR SIX) CARDS AND AN AMBIENT TABLE
64     C FOR EACH RUN. SETS OF CARDS CAN BE STACKED FOR MULTIPLE RUNS.
65     C

```

PROGRAM UDKHDEN

```

66 C
67 C          ***** INPUT VARIABLES *****
68 C
69 C          *** CARD NO. 1 *** FORMAT 20A4
70 C
71 C 1. THIS CARD CONTAINS ANY INFORMATION THE USER WISHES TO HAVE
72 C    PRINTED OUT AT THE TOP OF EACH PAGE. VARIABLE N11.
73 C
74 C    THE DATA ENTERED ON CARDS 2 THROUGH 7 MAY BE EITHER IN THE
75 C    FORMAT REQUIRED BY EACH CARD OR EACH VALUE ON THE CARD MAY
76 C    BE SEPARATED BY A COMMA (SHORT FIELD TERMINATION).
77 C
78 C          *** CARD NO. 2 *** FORMAT 3I2
79 C
80 C 1. INTER. IF INTER=0, ONLY ONE RUN OF THE PRESENT CASE IS
81 C    TO BE MADE. IF INTER=1, ONE OR MORE RUNS USING THE INPUT
82 C    DATA FOR THIS CASE ARE TO BE MADE WITH ONE OR MORE OF
83 C    THE INPUT VARIABLES CHANGED. THE USER WILL BE PROMPTED AT
84 C    THE END OF EACH RUN WHICH VARIABLES HE WISHES TO CHANGE.
85 C
86 C 2. IDFP. IF IDFP=1, THE INPUT FILE IS PRINTED AS PART OF
87 C    THE OUTPUT. IF IDFP=0, IT IS NOT PRINTED.
88 C
89 C 3. ICUTOP. NOT USED IN THIS PROGRAM. HOWEVER,
90 C    AS THIS IS A UNIVERSAL DATA FILE AND IF ICUTOP=1, THE
91 C    PROGRAM EXPECTS AN EXTRA CARD, I.E. CARD NO. 5.
92 C
93 C          *** CARD NO. 3 *** FORMAT F10.0,I10,3F10.0
94 C
95 C 1. QT=TOTAL DISCHARGE FLOW RATE (CUBIC METERS PER SEC)
96 C 2. NP=NUMBER OF DISCHARGE PORTS
97 C 3. PDIA=DISCHARGE PORT DIAMETER (M)
98 C 4. VANG=VERTICAL ANGLE (DEG) OF PORTS RELATIVE TO HORIZONTAL
99 C    (90 DEGREES IS VERTICAL, RANGE 0-130 DEG).
100 C 5. PDEP=PORT DEPTH (M) AND MUST BE GREATER THAN 0.0
101 C
102 C          *** CARD NO. 4 *** FORMAT 3F10.0
103 C
104 C 1. UN NOT USED IN THIS PROGRAM.
105 C 2. HANG=ANGLE (DEG) OF CURRENT DIRECTION WITH RESPECT TO DIFFUSER
106 C    AXIS (90 DEGREES CORRESPONDS TO A CURRENT HAVING A DIRECTION
107 C    PERPENDICULAR TO THE DIFFUSER AXIS, RANGE 45 - 135 DEG).
108 C 3. SPACE=DISTANCE BETWEEN ADJACENT PORTS (M) (IF NP=1 (CARD NO. 3),
109 C    SPACE=1000.0 DEFAULT)
110 C
111 C          *** CARD NO. 5 *** FORMAT F5.0,2I5,3I2,6F5.0,2I5
112 C
113 C THIS IS USED TO CHANGE PROGRAMMED DEFAULT VALUES IN THE OTHER
114 C PROGRAMS THAT USE THIS UDF. NO SUCH OPTIONS AVAILABLE IN THIS
115 C PROGRAM. IF ICUTOP=0 (CARD NO. 2), OMIT THIS CARD. IF ICUTOP=1,
116 C THIS CARD MUST BE INCLUDED AND MAY HAVE VALUES APPROPRIATE
117 C FOR THE OTHER PROGRAMS OR IT MAY BE BLANK. ZEROS OR BLANKS
118 C INDUCES THE OTHER PROGRAMS TO USE THEIR DEFAULT VALUES.
119 C
120 C          *** CARD NO. 6 *** FORMAT I10,2F10.0
121 C
122 C 1. NPTS=NUMBER OF ELEVATION ENTRIES IN AMBIENT PROFILE TABLE
123 C    (NPTS MUST BE AT LEAST 2 BUT NOT MORE THAN 30).
124 C 2. S=DISCHARGE SALINITY (PPT) IF TEMPERATURE/SALINITY OPTION USED
125 C    OR DISCHARGE DENSITY (G/CM3) IF DENSITY OPTION USED.
126 C 3. T=DISCHARGE TEMPERATURE (DEG C) IF TEMPERATURE/SALINITY
127 C    OPTION USED OR ZERO (0.0) IF DENSITY OPTION USED.
128 C
129 C          *** CARD NO. 7 *** FORMAT 4F10.0
130 C

```

PROGRAM UDKNDEN

```

131 C
132 C   THERE MUST BE NPTS (CARD NO. 6) IMAGES OF CARD NO. 7.
133 C   ONE CARD FOR EACH ELEVATION WHERE AMBIENT CONDITIONS
134 C   ARE GIVEN. AT LEAST 2 CARDS BUT NOT MORE THAN 30.
135 C 1. DP( )=DEPTH TO DATA POINT (N) STARTING AT THE SURFACE
136 C   AND WORKING DOWN TO OR BELOW THE DISCHARGE DEPTH.
137 C   THE LAST ENTRY MUST BE FOR A DEPTH EQUAL TO OR GREATER THAN
138 C   THE DISCHARGE DEPTH. MUST HAVE DATA FOR DP(1)=0.0 OR
139 C   COMPUTATIONAL ERRORS MAY OCCUR.
140 C 2. SA( )=AMBIENT SALINITY (PPT) AT THIS DEPTH IF THE
141 C   TEMPERATURE/SALINITY OPTION USED OR
142 C   AMBIENT DENSITY (G/CM3) IF DENSITY OPTION USED.
143 C 3. TA( )=AMBIENT TEMPERATURE (DEG C) AT THIS DEPTH IF TEMPERATURE/
144 C   SALINITY OPTION USED OR ZERO (0.0) IF DENSITY OPTION USED.
145 C 4. UA( )=AMBIENT VELOCITY AT THIS DEPTH (M/S)
146 C
147 C   REGARDING CARD NO 6 AND 7, WHEN USING THE TEMPERATURE/SALINITY
148 C   OPTION, IF FOR AN SA( ) OR S, THE TA( ) OR T IS REALLY ZERO,
149 C   GIVE TA( ) OR T A SMALL VALUE LIKE 0.000001.
150 C
151 C   THE ABOVE VARIABLE NAMES MATCH THE DESCRIPTION OF THE UNIVERSAL
152 C   DATA FILE. IN THIS PROGRAM SOME OF THEM ARE CHANGED SO BE
153 C   CAREFUL IF YOU MAKE ANY MODIFICATIONS.
154 C
155 C *****
156 C
157 C   IMPLICIT REAL*8(A-H,O-Z)
158 C   REAL*8 II1,II2,II3,II4
159 C   EXTERNAL DERIV,OUTP
160 C   LOGICAL IDEN
161 C   DIMENSION PRMT(5),AUX(16,6),F(6),FP(6),AA(10)
162 C   INTEGER*4 N11(20),N12(20)
163 C   COMMON/AMB2/XZ(30),XTA(30),XCA(30),XUA(30),DO,UO
164 C   COMMON/INPUT/ZZ(30),ZTA(30),ZCA(30),ZUA(30)
165 C   COMMON/AMB/CIO,TIO,PIO,RHOO,AL,TO,CO,FR,R1,FRL,TIS
166 C   COMMON/OUT/L,TH1,TH2,H,Z,SNEW,DS,K,N11,J2,J5,JJ,SPACE,SIGMA
167 C   *,DO,CD,AT,XXX,IOUT,ITERM
168 C   COMMON/PLTE/J6,J7,J8,J9,J10,X1H,Y1H,X2H,Y2H,Y3H
169 C   COMMON/ZP9/SET,XE,YE,ZE,BBE,DCE,DTE,RUE,RTE,RCE,T1E,T2E
170 C   COMMON/STR95/II,M,R3
171 C   COMMON/CONS/II1,II2,II3,II4,III
172 C   COMMON/WGM/WERGCK,ROVO
173 C   COMMON/OPT/IDEN
174 C   COMMON/TPDAT/ZTRAP,DILTP,PDEP,ITRAP,DILSF,ISURF,INTER
175 C   BYTE FNOUT(20),FNAME(20)
176 C   DATA FNOUT(20)/0/,FNAME(20)/0/
177 C
178 C   ASSIGN VALUES TO LOGICAL UNITS "IN" AND "IOUT"
179 C
180 C   IN THIS PROGRAM (PDP SYSTEM - FORTRAN IV-PLUS),
181 C   LOGICAL UNIT IN IS THE INPUT FILE.
182 C   " " IOUT IS THE OUTPUT FILE.
183 C   " " ITERM IS THE USER TERMINAL.
184 C
185 C   IN=1
186 C   IOUT=2
187 C   ITERM=5
188 C
189 C   OPEN FILES
190 C
191 C   WRITE(ITERM,554)
192 C 554 FORMAT(/1X,' *****PROGRAM UDKNDEN, AUGUST 1985*****')
193 C   WRITE(ITERM,3)
194 C   3 FORMAT(/1X,' ENTER UDF NAME: ',S)
195 C   READ(ITERM,4)(FNAME(JJ),JJ=1,19)

```

PROGRAM UDKNDEN

```

196      4 FORMAT(19A1)
197      WRITE(ITERM,5)
198      5 FORMAT(1X,' ENTER OUTPUT FILE NAME: ',S)
199      READ(ITERM,4)(FNOUT(JJ),JJ=1,19)
200      OPEN(UNIT=IN,NAME=FNAME,TYPE='OLD')
201      OPEN(UNIT=IOUT,NAME=FNOUT,TYPE='NEW')
202      C
203          XXX=0.16
204          A7=11.5
205          II1=.1285714
206          II2=.0667582
207          II3=0.45*1.07034
208          II4=0.31558*1.094
209          LOOP=0
210          NWO=0
211          NW=0
212      1 IF(L0OP.GT.0)GO TO 500
213          NWO=NWO+NW
214      C      N11 IS CASE IDENTIFICATION INFORMATION
215          READ(IN,102,END=221,ERR=999)N11
216      102 FORMAT(20A4)
217          READ(IN,8100,ERR=999)INTER,IDFP,ICUTOP
218      8100 FORMAT(3I2)
219          IF(INTER.EQ. 0)GO TO 500
220          WRITE(ITERM,7997)
221      C      N12 IS INTERACTIVE RUN TITLE FOR CASE N11
222          READ(ITERM,102)N12
223      500 CONTINUE
224      C
225      C      SET INITIAL VALUES
226      C
227          L=0
228          K=0
229          J7=0
230          J2=0
231          J9=0
232          J10=0
233          CIO=0.
234          TIO=0.
235          PIO=0.
236          AL=0.
237          R1=0.
238          Z=0.
239          SNEW=0.
240          R=0.
241          SIGMA=0.
242          X1M=0.
243          Y1M=0.
244          X2M=0.
245          Y2M=0.
246          Y3M=0.
247          SET=0.
248          XE=0.
249          YE=0.
250          ZE=0.
251          BBE=0.
252          DCE=0.
253          DTE=0.
254          RUE=0.
255          RTE=0.
256          RCE=0.
257          T1E=0.
258          T2E=0.
259          R3=0.
260          ROVO=0.

```

PROGRAM UDKHDEN

```

261      J5=0
262      CD=2.
263      DS=4.
264      JJ=1
265      J2=1
266      J6=0
267      II=1
268      SF=1000.
269      IF(LOOP.EQ.0)TIS=0.
270      III=1
271      ITRAP=0
272      ISURF=0
273      MERGCK=0
274      IF(LOOP.GT.0)GO TO 37
275      IDENE=0
276      IDENA=0
277      IDEN=.FALSE.
278  C
279      READ(IN,8101,ERR=999)QT,NP,PBIA,VANG,PDEP
280      8101  FORMAT(F10.0,I10,3F10.0)
281      READ(IN,8102,ERR=999)UW,HANG,HSPACE
282      8102  FORMAT(3F10.0)
283      IF(ICUTOP.GT.0)READ(IN,8104,ERR=999)IDUM4
284      8104  FORMAT(I1)
285      READ(IN,8103,ERR=999)NPTS,CO,TO
286      8103  FORMAT(I10,2F10.0)
287      TOE=TO
288      COE=CO
289      IF(TO.EQ.0.)IDENE=1
290  C
291  C      READ IN AMBIENT PROFILE TABLE
292  C      DEPTH BELOW SURFACE, DENSITY, AND VELOCITY
293  C
294      READ(IN,8105,ERR=999)(ZZ(II),ZCA(II),ZTA(II),ZUA(II),II=1,NPTS)
295      8105  FORMAT(4F10.0)
296      CALL USORT(ZZ,ZCA,ZTA,ZUA,NPTS)
297      NW=NPTS+5+ICUTOP
298      IF(IDFP.EQ.0)GO TO 42
299      WRITE(IOUT,15)(FNAME(JJ),JJ=1,19)
300      15  FORMAT(1H1/1X,' UNIVERSAL DATA FILE: ',19A1/)
301      IF(INTER.EQ.1)WRITE(IOUT,26)
302      26  FORMAT(
303      *1X,' *****'/
304      *1X,' *   NOTE, THIS IS THE ORIGINAL FILE.          */
305      *1X,' *   IT DOES NOT REFLECT CHANGES MADE INTERACTIVELY.  */
306      *1X,' *   THOSE CHANGES ARE SHOWN IN THE OUTPUT HEADING.    */
307      *1X,' *****'/)
308      REWIND IN
309      IF(NWO.EQ.0)GO TO 16
310      DO 17 JK=1,NWO
311      READ(IN,19)(AA(LK),LK=1,10)
312      19  FORMAT(10A8)
313      17  CONTINUE
314      DO 18 JK=1,NW
315      READ(IN,19)(AA(LK),LK=1,10)
316      WRITE(IOUT,22)(AA(LK),LK=1,10)
317      22  FORMAT(1X,10A8)
318      18  CONTINUE
319      42  CALL LIMITS(1,PDEP,VANG,HANG,TOE,COE,NPTS,INTER,N11,N12,
320      *IEXIT,IOUT,ITERM,TO,CO,LOOP,NP)
321      IF(IEXIT.EQ.1)GO TO 1
322      IF(ZTA(1).EQ.0.)IDENA=1
323      IF(IDENE.EQ.1.AND.IDENA.EQ.0)GO TO 11
324      IF(IDENE.EQ.0.AND.IDENA.EQ.1)GO TO 12
325      IF(IDENE.EQ.0.AND.IDENA.EQ.0)GO TO 13

```

PROGRAM UDKHDEH

```

326      IDEN=.TRUE.
327      GO TO 13
328      11 DO 14 J=1,NPTS
329          CALL SIGMAT(PI,ZTA(J),ZCA(J))
330          ZCA(J)=PI
331          ZTA(J)=0.
332      14 CONTINUE
333      IDEN=.TRUE.
334      GO TO 13
335      12 CALL SIGMAT(PI,TO,CO)
336      CO=PI
337      TO=0.
338      IDEN=.TRUE.
339      13 IF(IDEN)TO=CO
340      37 IF(IDEN)CO=.001
341          IF(CO.LT..001)CO=.001
342      XO=0.
343      YO=0.
344      ZO=0.
345      TH1=HANG
346      TH2=VANG
347      TH10=TH1
348      TH20=TH2
349  C      IF ONLY ONE DISCHARGE PORT, PORT SPACING IS SET
350  C      TO 1000. TO PREVENT PLUME MERGING CALCULATIONS.
351      IF(NP.EQ.1)HSPACE=1000.0
352      SPACE=HSPACE
353      DUO=1.
354      DRO=1.
355      DCO=1.
356      DTM=0.
357      WRITE(IOUT,200)
358      200 FORMAT(1H1,/42X,'PROGRAM UDKHDEH'/
359          *25X,'SOLUTION TO MULTIPLE BUOYANT DISCHARGE PROBLEM WITH'/
360          *25X,'AMBIENT CURRENTS AND VERTICAL GRADIENTS.  AUG 1985'/)
361      WRITE(IOUT,23)(FNAME(JJ),JJ=1,19)
362      23 FORMAT(4X,'UNIVERSAL DATA FILE: ',19A1)
363      WRITE(IOUT,24)N11
364      24 FORMAT(4X,'CASE I.D. ',20A4)
365      M=7
366      IF(INTER.EQ.1)WRITE(IOUT,25)N12
367      25 FORMAT(4X,'RUN TITLE: ',20A4)
368      IF(INTER.EQ.1)M=M+1
369      IF(NP.GT.1)GO TO 2003
370      WRITE(IOUT,2004)
371      2004 FORMAT(/10X,'SINGLE PORT DISCHARGE CASE')
372      M=M+1
373      2003 IF(IDEN)WRITE(IOUT,2000)QT,TO,PDIA,NP,HSPACE,PDEP
374      2000 FORMAT(1H,' DISCHARGE=',F7.4,' CU-M/S DENSITY='
375          *,F7.5,' G/CM3 ** DIAMETER=',F7.4,'-M',/1X,
376          *, '** NUMBER OF PORTS=',I4,' ** SPACING=',F7.2,'-M ** DEPTH
377          **',F7.2,'-M')
378      IF(.NOT.IDEN)WRITE(IOUT,2001)QT,TO,COE,PDIA,NP,HSPACE,PDEP
379      2001 FORMAT(1H,' DISCHARGE=',F7.4,' CU-M/S ** TEMPERATURE='
380          *,F6.2,'-C', '** SALINITY=',F6.3,'-PPT ** DIAMETER=',F7.4,
381          *,/1X, '** NUMBER OF PORTS=',I4,' ** SPACING=',F7.2,'-M
382          *, '** DEPTH=',F7.2,'-M')
383      M=M+2
384      DIL=1.
385      DO=PDIA
386      H=PDEP
387      ENN=NP
388      UO=QT/(ENN*3.1415926*DO*DO/4.)
389      IF(LOOP.GT.0)WRITE(IOUT,3200)
390      3200 FORMAT(/3X,' STRATIFICATION TABLE FROM PREVIOUS RUN USED'/)

```

PROGRAM UDKHDEH

```

391      IF(LOOP.GT.0)M=M+3
392      IF(LOOP.GT.0)GO TO 3001
393      IF(IDEN)WRITE(IOUT,3006)
394      IF(.NOT.IDEN)WRITE(IOUT,3009)
395      3006 FORMAT(1HD,10X,'AMBIENT STRATIFICATION PROFILE',/
396      * ,5X,'DEPTH (M)   DENSITY (G/CM3)   VELOCITY (M/S)')
397      3009 FORMAT(1HD,10X,'AMBIENT STRATIFICATION PROFILE',/
398      * ,7X,'DEPTH (M)   TEMP (C)   SALINITY (PPT)
399      *   DENSITY (G/CM3) VELOCITY (M/S)')
400      M=M+3
401      DO 400 II=1,NPTS
402      IF(IDEN)ZTA(II)=ZCA(II)
403      IF(IDEN)ZCA(II)=0.
404      IF(.NOT.IDEN.AND.TO.EQ.ZTA(II))TIS=1.
405      IF(.NOT.IDEN.AND.TO.EQ.ZTA(II))ZTA(II)=ZTA(II)-.05
406      IF(.NOT.IDEN)CALL SIGMAT(PI,ZTA(II),ZCA(II))
407      IF(IDEN)WRITE(IOUT,3007)ZC(II),ZTA(II),ZUA(II)
408      IF(.NOT.IDEN)WRITE(IOUT,3008)ZC(II),ZTA(II),ZCA(II),PI,ZUA(II)
409      3008 FORMAT(1X,F13.2,2F15.2,F15.5,F16.3)
410      3007 FORMAT(F13.2,F15.5,F16.3)
411      400 CONTINUE
412      M=M+NPTS
413      3001 CONTINUE
414      DO 4000 I=1,NPTS
415      J=NPTS-I+1
416      XZ(I)=H-ZZ(J)
417      XTA(I)=ZTA(J)
418      XCA(I)=ZCA(J)
419      XUA(I)=ZUA(J)
420      4000 CONTINUE
421      H=H/DO*2.
422      IF(IDEN)RHOO=TO
423      IF(.NOT.IDEN)CALL SIGMAT(RHOO,TO,CO)
424      DO=DO
425      Z=0.
426      CALL AMBIEN(Z,CIO,TIO,PIO,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
427      IF(TIO.EQ.1.)TIO=.999
428      ROVO=0.5*DO/UO
429      PID=ABS(PIO-1.)
430      FR=SQRT(PID*DO*9.8)
431      FR=UO/FR
432      IF(FR.GE.2.44)GO TO 99
433      WRITE(IOUT,8200)FR
434      IF(INTER .EQ. 1)WRITE(ITERM,8200)FR
435      8200 FORMAT(/3X,47(' ')/3X,'* THIS RUN DISCONTINUED, BECAUSE THE
436      * INITIAL *'/3X,'* FROUDE NUMBER (' ,F4.2,' ) IS LESS THAN 2.44,
437      * AND *'/3X,'* THE MODEL WILL NOT RUN CORRECTLY.',11X,'*'/
438      *3X,47(' ')/)
439      IF(INTER .EQ. 0)GO TO 1
440      GO TO 8005
441      99 SP=SF/DO*2.
442      SPACE=SPACE/DO
443      R1=R
444      RE=1.0 E+6
445      IF(ABS(PIZ)-1.0E-10)90,90,91
446      90 T=1.0 E+8
447      GO TO 92
448      91 T=(PIO-1.0)/(-PIZ)
449      92 WRITE(IOUT,206)FR,SPACE
450      206 FORMAT(1HD,' FROUDE NO=' ,F6.2,' ,',2X,
451      * 'PORT SPACING/PORT DIA=' ,F10.2,' ,',S)
452      M=M+2
453      FR=2.*FR*FR
454      SPACE=2.*SPACE
455      DS11=DS

```

PROGRAM UDKHDEN

```

456      TTT1=TH1
457      TTT2=TH2
458      R3=R*SIN(0.01745329*TH1)*COS(0.01745329*TH2)
459      IF(R3-1.0)60,61,60
460      61 R3=0.0
461      60 CALL ZFE(IHLF)
462      C ZFE TAKES CARE OF COMPUTING IC AT END OF ZFE.
463      IF(IHLF .GE. 11)GO TO 9
464      DS=DS11
465      TH1=TTT1*0.01745329
466      TH2=TTT2*0.01745329
467      T1E=T1E*0.01745329
468      T2E=T2E*0.01745329
469      C CONVERT TH FROM DEGREES TO RADIANS
470      C TH ANGLE AT S=0, TE=ANGLE AT SE
471      S1=SIN(T1E)
472      C2=COS(T2E)
473      CALL AMBIEN(Z,CI,TI,PI,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
474      RS12=R*S1*C2
475      IF(RE-1.0E+4)20,21,21
476      20 WRITE(IOUT,214)RE
477      214 FORMAT(1H , ' FLOW MAY NOT BE FULLY TURBULENT, RE=',F8.0,/)
478      M=M+1
479      21 PRMT(1)=SET
480      PRMT(2)=SF
481      PRMT(3)=DS
482      PRMT(4)=0.1
483      L=-1
484      N=6
485      C L=-1 USED TO INITIALIZE OUTP. N=NUMBER OF DIFFERENTIAL EQNS.
486      K=0
487      C K COUNTS NO OF INTEGRATIONS,
488      DO 2 I=1,6
489      2 FP(I)=0.1666667
490      SIGMA=BBE
491      BBE2=BBE*BBE
492      F(1)=BBE2*(II1+(.5-II1)*RS12)
493      F(2)=DTE*BBE2*(II2+(II1-II2)*RS12)
494      F(3)=DCE*BBE2*(II2+(II1-II2)*RS12)
495      F(4)=BBE2*II2*(1.-RS12)**2+2.*(1.-RS12)*RS12*BBE2*II1
496      **RS12*RS12*BBE2*.5
497      F(5)=T1E
498      F(6)=T2E
499      WRITE(IOUT,210)
500      210 FORMAT(/4X,'ALL LENGTHS ARE IN METERS-TIME IN SEC.
501      * FIRST LINE ARE INITIAL CONDITIONS.')


```

502 IF(IDEN)WRITE(IOUT,208)
503 IF(.NOT.IDEN)WRITE(IOUT,209)
504 208 FORMAT(1H ,5X,'X',7X,'Y',7X,'Z',6X,'TH1',5X,'TH2',5X,
505 *'WIDTH',5X,'DUCL',5X,'DRHO',5X,'DCCL',6X,'TIME',3X,'DILUTION'/
506 *)
507 209 FORMAT(1H ,5X,'X',7X,'Y',7X,'Z',6X,'TH1',5X,'TH2',5X,
508 *'WIDTH',5X,'DRHO',5X,'DTCL',5X,'DSCL',6X,'TIME',3X,'DILUTION'/
509 *)
510 WRITE(IOUT,4001)X0,Y0,Z0,TH10,TH20,D0,DU0,DRO,DCO,DTM,DIL
511 4001 FORMAT(1H ,5F8.2,1X,F8.2,3F9.3,1X,2F9.2)
512 M=M+5
513 CALL OUTP(PRMT(1),F,FP,IHLF,N,PRMT)
514 CALL HPCG(PRMT,F,FP,N,IHLF,DERIV,OUTP,AUX)
515 IF(IHLF .LT. 11)GO TO 27
516 9 WRITE(IOUT,10)IHLF
517 10 FORMAT(1X,' ERROR IN HPCG, IHLF= ',I2/
518 *1X,' SEE UNDER IHLF IN COMMENT SECTION OF SUBROUTINE HPCG,
519 * GOING TO NEXT DATA SET.')


```

520      IF(INTER .EQ. 1)WRITE(ITERM,10)IHLF

```


```


```

PROGRAM UDKHDEN

```

521      LOOP=0
522      GO TO 1
523 27 IF(ITRAP .EQ. 1)WRITE(IOUT,6)ZTRAP,DILTP
524 6 FORMAT(1X,' TRAPPING LEVEL=',F7.2,' METERS BELOW SURFACE,
525 * DILUTION=',F7.2)
526 IF(ISURF .EQ. 1 .AND. ITRAP .EQ. 0)WRITE(IOUT,8)DILSF
527 8 FORMAT(1X,' DILUTION=',F7.2)
528 IF(INTER.EQ.0)GO TO 1
529 IF(ITRAP .EQ. 1)WRITE(ITERM,6)ZTRAP,DILTP
530 IF(ISURF .EQ. 1 .AND. ITRAP .EQ. 0)WRITE(ITERM,8)DILSF
531 8005 WRITE(ITERM,8006)N11
532 8006 FORMAT(1X,' DO YOU WISH TO CHANGE ANY INPUT VARIABLES AND RERUN',
533 *1X,' CASE I.D. ',20A4/1X,' YES TO RERUN OR NO FOR NEXT CASE ',S)
534 READ(ITERM,7,ERR=999)MM
535 7 FORMAT(A1)
536 IF(MM.EQ.'N')LOOP=0
537 IF(MM.EQ.'N')GO TO 1
538 IF(MM.NE.'Y')WRITE(ITERM,201)
539 IF(MM.NE.'Y')GO TO 8005
540 201 FORMAT(1X,' *****YOU MADE A MISTAKE, TRY AGAIN*****')
541 WRITE(ITERM,7997)
542 7997 FORMAT(1X,' WHAT IS THE TITLE OF THIS INTERACTIVE RUN?'/2X,S)
543 READ(ITERM,102)M12
544 7999 IF(.NOT.IDEN)WRITE(ITERM,8000)QT,NP,PDIA,PDEP,VANG,
545 *HANG,COE,TO,HSPACE
546 IF(IDEN)WRITE(ITERM,8001)QT,NP,PDIA,PDEP,VANG,HANG,TO,HSPACE
547 8001 FORMAT(1X,' YOUR PRESENT INPUT VARIABLES ARE: '//
548 *1X,' 1. EFFLUENT FLOW (CMS) =',F15.4/
549 *1X,' 2. NUMBER OF DISCHARGE PORTS =',I10/
550 *1X,' 3. PORT DIAMETER (M) =',F15.4/
551 *1X,' 4. DISCHARGE DEPTH (M) =',F13.2/
552 *1X,' 5. VERTICAL DISCHARGE ANGLE =',F12.1/
553 *1X,' 6. ANGLE OF CURRENT RELATIVE TO DIFFUSER =',F6.1/
554 *1X,' 7. EFFLUENT DENSITY (G/CM3) =',F16.5/
555 *1X,' 8. PORT SPACING (M) =',F13.2//)
556 8000 FORMAT(1X,' YOUR PRESENT INPUT VARIABLES ARE: '//
557 *1X,' 1. EFFLUENT FLOW (CMS) =',F15.4/
558 *1X,' 2. NUMBER OF DISCHARGE PORTS =',I10/
559 *1X,' 3. PORT DIAMETER (M) =',F15.4/
560 *1X,' 4. DISCHARGE DEPTH (M) =',F13.2/
561 *1X,' 5. VERTICAL DISCHARGE ANGLE =',F12.1/
562 *1X,' 6. ANGLE OF CURRENT RELATIVE TO DIFFUSER =',F6.1/
563 *1X,' 7. EFFLUENT SALINITY (PPT) =',F14.3/
564 *1X,' 8. EFFLUENT TEMPERATURE (C) =',F13.2/
565 *1X,' 9. PORT SPACING (M) =',F13.2//)
566 WRITE(ITERM,7998)
567 7998 FORMAT(1X,' ENTER THE NUMBER OF THE
568 * VARIABLE YOU WISH TO CHANGE: ',S)
569 READ(ITERM,8106)INUM
570 8106 FORMAT(I1)
571 WRITE(ITERM,7996)
572 7996 FORMAT(1X,' WHAT IS THE NEW VALUE OF THIS VARIABLE? ',S)
573 READ(ITERM,8107,ERR=7999)VALUE
574 8107 FORMAT(F10.0)
575 IF(INUM.EQ.1)QT=VALUE
576 IF(INUM.EQ.2)NP=VALUE
577 IF(INUM.EQ.3)PDIA=VALUE
578 IF(INUM.EQ.4)PDEP=VALUE
579 IF(INUM.EQ.5)VANG=VALUE
580 IF(INUM.EQ.6)HANG=VALUE
581 IF(.NOT.IDEN.AND.INUM.EQ.7)CO=VALUE
582 IF(.NOT.IDEN.AND.INUM.EQ.7)COE=CO
583 IF(.NOT.IDEN.AND.INUM.EQ.8)TO=VALUE
584 IF(.NOT.IDEN.AND.INUM.EQ.8)TOE=TO
585 IF(IDEN.AND.INUM.EQ.7)GO TO 501

```

PROGRAM UDKHDEH

```

586 502 IF(IDEN.AND.INUM.EQ.8)HSPACE=VALUE
587 IF(.NOT.IDEN.AND.INUM.EQ.9)HSPACE=VALUE
588 202 WRITE(ITERM,7995)
589 7995 FORMAT(1X,' DO YOU WISH TO CHANGE ANOTHER VARIABLE?
590 *YES OR NO: ',S)
591 READ(ITERM,7)MM
592 IF(MM.EQ.'Y')GO TO 7999
593 IF(MM.NE.'N')WRITE(ITERM,201)
594 IF(MM.NE.'N')GO TO 202
595 LOOP=1
596 CALL LIMITS(2,PDEP,VANG,HANG,TOE,COE,NPTS,INTER,N11,N12,
597 *IEXIT,IOUT,ITERM,TO,CO,LOOP,NP)
598 IF(IEXIT.EQ.1)LOOP=0
599 GO TO 1
600 501 TO=VALUE
601 TOE=0.
602 COE=TO
603 GO TO 502
604 999 WRITE(IOUT,9999)
605 9999 FORMAT(1X,' ERROR IN INPUT DATA. PLEASE CHECK FILE')
606 221 CALL EXIT
607 END
608 C
609 SUBROUTINE LIMITS(KK,PDEP,VANG,HANG,TOE,COE,NPTS,INTER,N11,N12,
610 *IEXIT,IOUT,ITERM,TO,CO,LOOP,NP)
611 C
612 C THIS SUBROUTINE CHECKS LIMITS OF PORT DEPTH (GREATER THAN 0. AND LESS
613 C THAN OR EQUAL TO PROFILE DEPTH), DISCHARGE ANGLE (EQUAL TO OR GREATER
614 C THAN -5 DEG BUT LESS THAN OR EQUAL TO 130 DEG, AND IF NEGATIVE, THAT
615 C THERE IS AT LEAST ONE PROFILE DATA POINT ONE METER OR MORE BELOW THE
616 C PORT DEPTH.) (COMPUTATIONAL PROBLEMS MAY OCCUR WITH LARGER DISCHARGE
617 C ANGLES, I.E. SQUARE ROOT OF A NEGATIVE NUMBER, AND PROGRAM WOULD
618 C EXIT), AND CURRENT ANGLE RELATIVE TO THE DIFFUSER (EQUAL TO OR GREATER
619 C THAN 45 BUT LESS THAN OR EQUAL TO 135 DEG. 90 DEG IS PERPENDICULAR TO
620 C THE DIFFUSER). HOWEVER, IF ITS A SINGLE PORT DISCHARGE ANY CURRENT
621 C ANGLE IN THE RANGE OF 0 TO 180 IS OK, BUT FOR ANGLES GREATER THAN 90
622 C DEG THE PROGRAM SETS THE CURRENT ANGLE (HANG) EQUAL TO THE
623 C SUPPLEMENTARY ANGLE. THAT THE EFFLUENT DENSITY IS LESS THAN THE
624 C AMBIENT DENSITY AND THAT THERE ARE AMBIENT PROFILE VALUES FOR THE
625 C SURFACE, I.E. DP(1)=0. IF INTER=1, THE PORT DEPTH, ANGLES AND EFFLUENT
626 C DENSITY CAN BE CORRECTED INTERACTIVELY BUT SURFACE DATA CORRECTION
627 C MUST BE MADE TO THE DATA SET(S) AND REENTERED. IF INTER=0, ALL
628 C CORRECTION MUST BE MADE TO THE DATA SET(S) AND REENTERED.
629 C
630 IMPLICIT REAL*8(A-H,O-Z)
631 COMMON/INPUT/ZZ(30),ZTA(30),ZCA(30),ZUA(30)
632 INTEGER*4 N11(20),N12(20)
633 LIMIT=0
634 IDEEP=0
635 IVANG=0
636 NEGANG=0
637 IHANG=0
638 IHANG1=0
639 IRHO=0
640 INO=0
641 IEXIT=0
642 100 IF(PDEP.EQ.0..OR.ZZ(NPTS).LT.PDEP)GO TO 1
643 10 IF(VANG.LT.0..OR.VANG.GT.130.)GO TO 101
644 110 IF(NP.GT.1)GO TO 20
645 IF(HANG.GT.90.)HANG=180.-HANG
646 IF(HANG.LT.0..OR.HANG.GT.90.)GO TO 111
647 GO TO 30
648 20 CONTINUE
649 IF(HANG.LT.45..OR.HANG.GT.135.)GO TO 21
650 30 CONTINUE

```

PROGRAM UDKNDEN

```

651      IF(TOE .EQ. 0.)RHOE=COE
652      IF(TOE .NE. 0.)CALL SIGMAT(RHOE,TOE,COE)
653      DO 31 J=2,NPTS
654      IF(PDEP .LE. ZZ(J))GO TO 32
655 31 CONTINUE
656      GO TO 40
657 32 JK=J
658      PRO=(PDEP-ZZ(JK-1))/(ZZ(JK)-ZZ(JK-1))
659      TT=ZTA(JK-1)+PRO*(ZTA(JK)-ZTA(JK-1))
660      SS=ZCA(JK-1)+PRO*(ZCA(JK)-ZCA(JK-1))
661      IF(KK .EQ. 2)GO TO 38
662      IF(ZTA(JK) .EQ. 0.)RHOA=SS
663      IF(ZTA(JK) .NE. 0.)CALL SIGMAT(RHOA,TT,SS)
664      GO TO 39
665 38 IF(ZCA(JK) .EQ. 0.)RHOA=TT
666      IF(ZCA(JK) .NE. 0.)CALL SIGMAT(RHOA,TT,SS)
667 39 IF(RHOA .GT. RHOE)GO TO 40
668      IRHO=1
669      LIMIT=1
670      IF(INTER .EQ. 0)GO TO 40
671 120 CONTINUE
672      WRITE(ITEM,33)RHOE,RHOA
673 33 FORMAT(/1X,' EFFLUENT DENSITY (' ,F7.5,' G/CM3) MUST BE'/
674      *1X,' LESS THAN THE AMBIENT DENSITY (' ,F7.5,' G/CM3)')
675      WRITE(ITEM,4)
676      READ(ITEM,5)NANS
677      IF(NANS .EQ. 'N')GO TO 40
678      IF(NANS .NE. 'Y')WRITE(ITEM,200)
679      IF(NANS .NE. 'Y')GO TO 120
680      IRHO=2
681      IF(TOE .EQ. 0.)GO TO 36
682      WRITE(ITEM,34)
683 34 FORMAT(/1X,' NEW EFFLUENT TEMP VALUE? ' ,S)
684      READ(ITEM,7)TO
685      TOE=TO
686      WRITE(ITEM,35)
687 35 FORMAT(1X,' NEW EFFLUENT SAL VALUE? ' ,S)
688      READ(ITEM,7)CO
689      COE=CO
690      GO TO 30
691 36 WRITE(ITEM,37)
692 37 FORMAT(/1X,' NEW EFFLUENT DENSITY VALUE? ' ,S)
693      READ(ITEM,7)CO
694      IF(KK .EQ. 2)TO=CO
695      COE=CO
696      GO TO 30
697 101 IF(VANG .LT. -5. .OR. VANG .GT. 130.)GO TO 11
698 C
699 C      IF CONTROL IS NOT PASSED TO 11, VANG .GE. -5. BUT .LT. 0.
700 C      BECAUSE IT'S NEGATIVE, PROFILE DEPTH MUST BE AT LEAST ONE
701 C      METER DEEPER THAN THE PORT DEPTH.
702 C
703      IF(ZZ(NPTS)-PDEP .GE. 1.0)GO TO 110
704      NEGANG=1
705 C
706      1 IDEEP=1
707      LIMIT=1
708      IF(INTER .EQ. 0)GO TO 10
709 121 CONTINUE
710      IF(PDEP .EQ. 0.)WRITE(ITEM,2)
711 2 FORMAT(/1X,' ZERO DISCHARGE DEPTH NOT ALLOWED,
712      * CHANGE PORT DEPTH.')
713      IF(PDEP .GT. 0. .AND. ZZ(NPTS) .LT. PDEP)
714      *WRITE(ITEM,3)PDEP,ZZ(NPTS)
715 3 FORMAT(/1X,' PORT DEPTH (' ,F6.2,' M) IS DEEPER THAN'/

```

PROGRAM UDKNDEN

```

716      *1X,' THE PROFILE DEPTH ('F6.2,' M), CHANGE PORT DEPTH.')
717      IF(NEGANG .EQ. 1)WRITE(ITERM,23)PDEP,ZZ(NPTS)
718 23  FORMAT(/1X,' FOR A NEGATIVE DISCHARGE ANGLE, THE PORT DEPTH ('
719      'F6.4,' M) MUST BE AT'1X,' LEAST ONE METER LESS THAN THE PROFILE
720      ' DEPTH ('F6.4,' M), CHANGE PORT DEPTH.')
721      WRITE(ITERM,4)
722      4  FORMAT(/1X,' WANT TO MAKE CORRECTION? YES/NO ',S)
723      READ(ITERM,5)NANS
724      5  FORMAT(A1)
725      IF(NANS .EQ. 'N')GO TO 10
726      IF(NANS .NE. 'Y')WRITE(ITERM,200)
727      IF(NANS .NE. 'Y')GO TO 121
728      IDEEP=2
729      WRITE(ITERM,6)
730      6  FORMAT(1X,' NEW VALUE? ',S)
731      READ(ITERM,7)PDEP
732      7  FORMAT(F7.0)
733      GO TO 100
734 11  IVANG=1
735      LIMIT=1
736      IF(INTER .EQ. 0)GO TO 110
737 122 CONTINUE
738      WRITE(ITERM,12)
739      12 FORMAT(/1X,' DISCHARGE ANGLE IS NOT IN THE
740      * RANGE OF -5 TO 130 DEG')
741      WRITE(ITERM,4)
742      READ(ITERM,5)NANS
743      IF(NANS .EQ. 'N')GO TO 110
744      IF(NANS .NE. 'Y')WRITE(ITERM,200)
745      IF(NANS .NE. 'Y')GO TO 122
746      IVANG=2
747      WRITE(ITERM,6)
748      READ(ITERM,7)VANG
749      GO TO 10
750 111 IHANG1=1
751      LIMIT=1
752      IF(INTER .EQ. 0)GO TO 20
753 123 CONTINUE
754      WRITE(ITERM,112)
755 112 FORMAT(/
756      *1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER IS NOT IN THE'
757      *1X,' RANGE OF 0 TO 180 DEG FOR A SINGLE PORT DISCHARGE. IF'
758      *1X,' INPUT ANGLE .GT. 90. PROGRAM SETS HANG=SUPPLEMENTARY ANGLE.')
759      WRITE(ITERM,4)
760      READ(ITERM,5)NANS
761      IF(NANS .EQ. 'N')GO TO 20
762      IF(NANS .NE. 'Y')WRITE(ITERM,200)
763      IF(NANS .NE. 'Y')GO TO 123
764      IHANG1=2
765      WRITE(ITERM,6)
766      READ(ITERM,7)HANG
767      GO TO 110
768 21  IHANG=1
769      LIMIT=1
770      IF(INTER .EQ. 0)GO TO 30
771 124 CONTINUE
772      WRITE(ITERM,22)
773      22 FORMAT(/1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER
774      * IS NOT IN THE RANGE OF 45 TO 135 DEG')
775      WRITE(ITERM,4)
776      READ(ITERM,5)NANS
777      IF(NANS .EQ. 'N')GO TO 30
778      IF(NANS .NE. 'Y')WRITE(ITERM,200)
779      IF(NANS .NE. 'Y')GO TO 124
780      IHANG=2

```

PROGRAM UBKHDEN

```

781      WRITE(ITERM,6)
782      READ(ITERM,7)HANG
783      GO TO 20
784      40 IF(ZZ(1) .EQ. 0.)GO TO 50
785      INO=1
786      LIMIT=1
787      IF(INTER .EQ. 0)GO TO 50
788      WRITE(ITERM,41)N11
789      41 FORMAT(/1X,' COMPUTATIONS CEASE FOR'/1X,' CASE I.D. ',20A4)
790      WRITE(ITERM,44)N12
791      44 FORMAT(1X,' MUN TITLE: ',20A4)
792      WRITE(ITERM,42)
793      42 FORMAT(/1X,' NO SURFACE DATA FOR THE AMBIENT DENSITY/CURRENT
794      * PROFILE.'/1X,' WITHOUT THESE DATA, COMPUTATIONAL ERRORS
795      * MAY OCCUR.'/1X,' MAKE CORRECTIONS TO DATA SET(S) AND REENTER.')
```

```

796      WRITE(ITERM,43)
797      43 FORMAT(/1X,' GOING TO NEXT DATA SET IF THERE IS ONE.')
```

```

798      50 IF(LIMIT .EQ. 0)GO TO 70
799      IF(INTER .EQ. 1)GO TO 69
800      51 IF(LOOP .EQ. 1)WRITE(IOUT,72)
801      72 FORMAT(1H1)
802      WRITE(IOUT,41)N11
803      IF(INTER .EQ. 0)GO TO 73
804      WRITE(IOUT,44)N12
805      WRITE(ITERM,41)N11
806      WRITE(ITERM,44)N12
807      WRITE(ITERM,74)
808      74 FORMAT(/1X,' NECESSARY CORRECTIONS NOT MADE!')
```

```

809      WRITE(ITERM,43)
810      73 WRITE(IOUT,67)
811      67 FORMAT(/1X,' CORRECT THE FOLLOWING AND REENTER DATA.')
```

```

812      IEX.T=1
813      68 IF(IDEEP .EQ. 0)GO TO 52
814      IF(IDEEP .EQ. 1)WRITE(IOUT,56)PDEP
815      IF(IDEEP .EQ. 2)WRITE(IOUT,57)PDEP
816      52 IF(IVANG .EQ. 0)GO TO 520
817      IF(IVANG .EQ. 1)WRITE(IOUT,58)
818      IF(IVANG .EQ. 2)WRITE(IOUT,59)VANG
819      520 IF(IHANG1 .EQ. 0)GO TO 53
820      IF(IHANG1 .EQ. 1)WRITE(IOUT,521)
821      IF(IHANG1 .EQ. 2)WRITE(IOUT,63)HANG
822      53 IF(IHANG .EQ. 0)GO TO 54
823      IF(IHANG .EQ. 1)WRITE(IOUT,62)
824      IF(IHANG .EQ. 2)WRITE(IOUT,63)HANG
825      54 IF(IRHO .EQ. 0)GO TO 55
826      IF(IRHO .EQ. 1)WRITE(IOUT,64)
827      IF(IRHO .EQ. 2)WRITE(IOUT,65)RHOE
828      55 IF(INO .EQ. 0)GO TO 75
829      WRITE(IOUT,42)
830      WRITE(IOUT,43)
831      GO TO 70
832      69 IF(IDEEP .EQ. 1 .OR. IVANG .EQ. 1 .OR. IHANG1 .EQ. 1
833      * .OR. IHANG .EQ. 1 .OR. IRHO .EQ. 1 .OR. INO .EQ. 1)GO TO 51
834      IF(LOOP .EQ. 1)WRITE(IOUT,72)
835      WRITE(IOUT,66)N11
836      IF(INTER .EQ. 1)WRITE(IOUT,44)N12
837      WRITE(IOUT,71)
838      GO TO 68
839      56 FORMAT(
840      *1X,' PORT DEPTH WAS ENTERED AS ',F6.2,' METERS:'/
841      *1X,' IF THE DISCHARGE ANGLE IS POSITIVE, THE PORT DEPTH MUST'/
842      *1X,' BE GREATER THAN ZERO BUT LESS THAN OR EQUAL TO THE PROFILE
843      * DEPTH. '/
844      *1X,' IF THE DISCHARGE ANGLE IS NEGATIVE, THE PORT DEPTH MUST'/
845      *1X,' BE AT LEAST ONE METER LESS THAN THE PROFILE DEPTH.')
```

PROGRAM UDKHDEH

```

846 57 FORMAT(1X,' PORT DEPTH CHANGED TO: ',F6.2,' M')
847 58 FORMAT(1X,' DISCHARGE ANGLE MUST BE .GE. -5 BUT .LE. 135 DEG')
848 59 FORMAT(1X,' DISCHARGE ANGLE CHANGED TO: ',F5.1,' DEG')
849 521 FORMAT(1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER
850 * MUST BE .GE. 0 BUT .LE. 180 DEG'/1X,
851 * ' FOR A SINGLE PORT DISCHARGE')
852 62 FORMAT(1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER
853 * MUST BE .GE. 45 BUT .LE. 135 DEG')
854 63 FORMAT(1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER
855 * CHANGED TO: ',F6.2,' DEG')
856 64 FORMAT(1X,' EFFLUENT DENSITY MUST BE .LE. AMBIENT
857 * DENSITY AT THE DISCHARGE DEPTH')
858 65 FORMAT(1X,' EFFLUENT DENSITY CHANGED TO: ',F7.5,' G/CC')
859 66 FORMAT(1X,' COMPUTATIONS CONTINUE FOR'/1X,' CASE I.D. ',20A4)
860 71 FORMAT(1X,' CORRECTIONS WERE INTERACTIVELY MADE TO
861 * THE FOLLOWING: '/')
862 200 FORMAT(1X,' *****YOU MADE A MISTAKE, TRY AGAIN*****')
863 75 IF(IEXIT .EQ. 1)WRITE(IOUT,43)
864 70 RETURN
865 END
866 C
867 SUBROUTINE SIGMAT(RHO,T,S)
868 IMPLICIT REAL*8(A-H,O-Z)
869 SIGO=((6.8E-6*S)-4.82E-4)*S+.8149)*S-.093
870 C=1.E-6*T*((.01667*T-.0164)*T+18.03)
871 D=.001*T*((.0010843*T-.09818)*T+4.7367)
872 SUHT=(T-3.98)*(T-3.98)*(T+283.)/(503.57*(T+67.26))
873 RHO=(SIGO+.1324)*(1.-D+C*(SIGO-.1324))-SUHT
874 RHO=RHO/1000.+1.
875 RETURN
876 END
877 C
878 SUBROUTINE DERIV(S,F,FP)
879 IMPLICIT REAL*8(A-H,O-Z)
880 DIMENSION F(6),FP(6)
881 INTEGER*4 N11(20)
882 COMMON/OUT/L,TH1,TH2,H,Z,SOLD,DS,X,N11,J2,J5,JJ,SPACE,SIGMA
883 *,DO,CD,A7,XXX,IOUT,ITERM
884 S1=SIN(F(5))
885 S2=SIN(F(6))
886 C1=COS(F(5))
887 C2=COS(F(6))
888 Z1=Z+S2*(S-SOLD)
889 CALL AMBIEN(Z1,C1,TI,PI,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
890 C AMBIEN COMPUTES GRADIENTS OF T,C & TURBULENCE QUANTITIES IN FR STREAM.
891 CALL PRFLE(F,F8,E,B)
892 C PRFLE COMPUTES INTEGRAL QUANTITIES WITHIN THE JET
893 C F8 IS THE INTEGRAL OF DENSITY DIFFERENCE, AND B THE WIDTH OF TE JET
894 FP(1)=E
895 FP(2)=-F(1)*TIZ+S2+VTI
896 FP(3)=-F(1)*CIZ+S2+VCI
897 FP(4)=E*R+S1*C2+F8+S2+VUI
898 P9=F(4)-0.25*E+E+VVI
899 IF(ABS(C2)-0.01)1,1,2
900 1 F(5)=F(6)
901 FP(5)=0.0
902 GO TO 3
903 2 F(5)=0.
904 UNX=-(C2**2+C1*S1)
905 UNY=S2**2+C2**2+C1**2
906 UNZ=-(S2*C2*S1)
907 UN=SQRT(UNX**2+UNY**2+UNZ**2)
908 CDP=CD*SPACE/B
909 IF(SPACE/B.GE.2.) CDP=CD*B/SPACE
910 FD=CDP*B*R*R*UN/6.2832

```

PROGRAM UDKHDEN

```

911      YY=ABS(C1+C2)
912      IF(YY.GT.0.)FDY=FD+(S2**2+C2**2+C1**2-S1**2+S2**2)/(F9+C1+C2)
913      FP(S)=(E+R+C1)/(C2+F9)+FDY
914      3 FP(6)=(F8+C2-(E+R+FD)*S1+S2)/F9
915      K=K+1
916      RETURN
917      END
918  C
919      SUBROUTINE XINT(I,A,F1INT,F2INT,F3INT)
920      IMPLICIT REAL*8(A-H,O-Z)
921      DIMENSION XA(13),X11(12),X12(12)
922      DATA XA/1.0E-10,1.,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9,
923      *2.0,100./
924      DATA X11/.4314,.4314,.4285,.4259,.4238,.4221,.4209,
925      *.4201,.4196,.4194,.4193,.4193/
926      DATA X12/.436,.3905,.3576,
927      *.3348,.3201,.3114,.3068,.3047,.304,.3038,.30377,.30377/
928      1 IF(A-XA(I))2,3,4
929      2 I=I-1
930      GO TO 1
931      3 F1INT=X11(I)
932      F2INT=X12(I)
933      GO TO 8
934      4 IF(A-XA(I+1))5,6,7
935      6 F1INT=X11(I+1)
936      F2INT=X12(I+1)
937      I=I+1
938      GO TO 8
939      7 I=I+1
940      GO TO 4
941      5 F1INT=X11(I)+(X11(I+1)-X11(I))/(XA(I+1)-XA(I))*(A-XA(I))
942      F2INT=X12(I)+(X12(I+1)-X12(I))/(XA(I+1)-XA(I))*(A-XA(I))
943      8 F3INT=A*SQRT(1.-(A/2.)**2)/2.+ASIN(A/2.)
944      RETURN
945      END
946  C
947      SUBROUTINE USORT(DP,X,Y,Z,NPTS)
948      IMPLICIT REAL*8(A-H,O-Z)
949      DIMENSION DP(30),X(30),Y(30),Z(30)
950      NESTED=NPTS
951      L=NESTED-1
952      DO 1000 N=1,L
953          NESTED=NESTED-1
954          DO 1000 I=1,NESTED
955              IF(DP(I).LE.DP(I+1))GO TO 1000
956              DUMMY=DP(I)
957              DP(I)=DP(I+1)
958              DP(I+1)=DUMMY
959              DUMMY=X(I)
960              X(I)=X(I+1)
961              X(I+1)=DUMMY
962              DUMMY=Y(I)
963              Y(I)=Y(I+1)
964              Y(I+1)=DUMMY
965              DUMMY=Z(I)
966              Z(I)=Z(I+1)
967              Z(I+1)=DUMMY
968      1000 CONTINUE
969      RETURN
970      END
971  C
972      SUBROUTINE AMBIEN(Z,C1,T1,P1,T1Z,C1Z,P1Z,VTI,VCI,VUI,VVI,R)
973  C AMBIEN COMPUTES THE VALUES OF PARAMETERS IN THE FREE STREAM
974  C LOCAL DENSITY, VELOCITY AND THEIR GRADIENTS
975  C

```

PROGRAM UDKHDEN

```

976      IMPLICIT REAL*8(A-H,O-Z)
977      COMMON/AMB2/XZ(30),XTA(30),XCA(30),XUA(30),DO,UO
978      COMMON/AMB/CIO,TIO,PIO,RHOO,AL,TO,CO,FR,R1,FRL,TIS
979      COMMON/TPDAT/ZTRAP,DILTP,PDEP,ITRAP,DILSF,ISURF,INTER
980      COMMON/OPT/IDEN
981      LOGICAL IDEN
982      DO=DO
983      I=1
984      Z1=Z+DO/2.
985      IF(Z1 .GE. PDEP) Z1=PDEP
986      1 IF(Z1.GT.XZ(I+1))GO TO 2
987      TIZ=(XTA(I+1)-XTA(I))/(XZ(I+1)-XZ(I))
988      CIZ=(XCA(I+1)-XCA(I))/(XZ(I+1)-XZ(I))
989      UIZ=(XUA(I+1)-XUA(I))/(XZ(I+1)-XZ(I))
990      CI=XCA(I)+CIZ*(Z1-XZ(I))
991      TI=XTA(I)+TIZ*(Z1-XZ(I))
992      R=(XUA(I)+UIZ*(Z1-XZ(I)))/UO
993      IF(.NOT.IDEN)CALL SIGMAT(PI,TI,CI)
994      IF(.NOT.IDEN)CALL SIGMAT(PI1,XTA(I),XCA(I))
995      IF(.NOT.IDEN)CALL SIGMAT(PI2,XTA(I+1),XCA(I+1))
996      IF(.NOT.IDEN)PIZ=(PI2-PI1)*DO/(2.*RHOO*(XZ(I+1)-XZ(I)))
997      TI=TI/TO
998      IF(.NOT.IDEN)CI=CI/CO
999      IF(.NOT.IDEN)PI=PI/RHOO
1000     TIZ=TIZ*DO/(2.*TO)
1001     IF(.NOT.IDEN)CIZ=CIZ*DO/(2.*CO)
1002     IF(IDEN)PI=TI
1003     IF(IDEN)PIZ=TIZ
1004     C
1005     C VTI, VCI, VUI, AND VVI ARE AMBIENT TURBULENCE TERMS, PRESENTLY
1006     C NOT USED AS THEY ARE NOT IMPORTANT IN THE NEAR FIELD. THUS,
1007     C THEY ARE SET EQUAL TO ZERO.
1008     C
1009     VTI=0.
1010     VCI=0.
1011     VUI=0.
1012     VVI=0.
1013     C
1014     RETURN
1015     2 I=I+1
1016     GO TO 1
1017     END
1018     C
1019     SUBROUTINE PRFLE(F,FB,E,B)
1020     IMPLICIT REAL*8(A-H,O-Z)
1021     REAL*8 I11,I12,I13,I14
1022     DIMENSION F(6)
1023     INTEGER*4 N11(20)
1024     COMMON/CONS/I11,I12,I13,I14,I11
1025     COMMON/AMB/CIO,TIO,PIO,RHOO,AL,TO,CO,FR,R1,FRL,TIS
1026     COMMON/OUT/L,TH1,TH2,H,Z,SNEW,DS,K,N11,J2,J5,JJ,SPACE,SIGMA
1027     *,DO,CD,A7,XXX,IOUT,ITERM
1028     COMMON/OPT/IDEN
1029     LOGICAL IDEN
1030     I=I11
1031     PI=3.14159
1032     IF(F(4))12,12,11
1033     11 S12=SIN(F(5))*COS(F(6))
1034     CALL AMBIEN(Z,CI,TI,RHO,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
1035     U12=R*S12
1036     C **** SINGLE PLUME ****
1037     A=SPACE/SIGMA
1038     IF(A-2.)501,500,500
1039     501 IF(A-.95)503,503,502
1040     500 C1=I11

```

PROGRAM UDKNDEN

```

1041      C2=U12/2.
1042      C3=I12
1043      C4=2.*U12*I11
1044      C5=U12*U12/2.
1045      C6=I12
1046      C7=U12*I11
1047      GO TO 504
1048 C **** MERGING PLUME ****
1049 502 CALL XINT(I,A,AI1,AI2,AI3)
1050      C1=2.*I13*AI1/PI
1051      C2=U12*AI3/PI
1052      C3=2.*I14*AI2/PI
1053      C4=4.*I13*AI1*U12/PI
1054      C5=U12*U12*AI3/PI
1055      C6=2.*I14*AI2/PI
1056      C7=2.*I13*U12*AI1/PI
1057      GO TO 504
1058 C **** MERGED PLUME ****
1059 503 CALL XINT(I,A,AI1,AI2,AI3)
1060      C1=.45*AI3/PI
1061      C2=U12*AI3/PI
1062      C3=.31558*AI3/PI
1063      C4=2.*U12*C1
1064      C5=U12*C2
1065      C6=C3
1066      C7=0.5*C4
1067 504 IF (ABS(U12)-.001)112,112,113
1068 112 BSQR=F(1)*F(1)*C3/(F(4)*C1*C1)
1069      GO TO 12
1070 113 AA=C3*C2*C2/C1/C1-C4*C2/C1*C5
1071      BB=C4*F(1)/C1-C3*2.*C2*F(1)/C1/C1-F(4)
1072      CC=C3*F(1)*F(1)/C1/C1
1073      RADCL=BB*BB-4.*AA*CC
1074      IF (RADCL.LE.0.)RADCL=0.
1075      BSQR=-BB-SQRT(RADCL)
1076      BSQR=BSQR/(2.*AA)
1077 12 B=SQRT(BSQR)
1078      SIGMA=B
1079      DLUM=F(1)/(BSQR*C1)-C2/C1
1080      UCL=DLUM*U12
1081      DTCL=F(2)/(BSQR*(DLUM*C6+C7))
1082      DCCL=F(3)/(BSQR*(DLUM*C6+C7))
1083      TCL=(DTCL+TI)*TO
1084      IF (.NOT.IDEN)CCL=(DCCL+CI)*CO
1085      IF (.NOT.IDEN)CALL SIGMAT(PICL,TCL,CCL)
1086      IF (.NOT.IDEN)PICL=RHO-PICL/RHO0
1087      IF (IDEN)PICL=-DTCL
1088      IF (FR-0.98E+6)1,2,2
1089 1 IF (A.GT.2.)FB=PICL*BSQR*I11/((PIO-1.)*FR)
1090      IF (A.LT.2.)FB=2.*BSQR*I13*PICL*AI1/(PI*FR*(PIO-1.))
1091      IF (A.LE..95)FB=FB*AI3/AI1/2.14068
1092 C COMPUTE FB-INTEGRAL OF DENSITY DIFFERENCE.
1093      FRLI=PICL*B/(FR*(PIO-1.0)*UCL*UCL)
1094      GO TO 3
1095 2 FB=0.0
1096      FRLI=0.0
1097 3 IF (ABS(FRLI)-10.0*ABS(1.0/FRLI))10,13,13
1098 13 FRLI=1.0/FRLI
1099 10 CONTINUE
1100      A1=.05
1101      A22=0.0
1102      AA=A1+A22*FRLI*SIN(F(6))
1103      CC=A7*R*SQRT(1.0-S12*S12)
1104      BB=B*ABS(UCL-R*S12)
1105      IF (A.GE.2.)E=AA*(BB*(1.-XXX/A)+CC*B)

```

PROGRAM UDKNDEN

```

1106 IF(A.LT.2.)E=AA*(BB*(1.-XXX/2.)*(1.-2./PI*ACOS(A/2.))
1107 **CC=SPACE/2.)
1108 III=I
1109 C COMPUTE E-ENTRAINMENT FUNCTION
1110 RETURN
1111 END
1112 C
1113 SUBROUTINE OUTP(S,F,FP,IN,N,PRNT)
1114 IMPLICIT REAL*8(A-H,O-Z)
1115 REAL*8 II1,II2,II3,II4
1116 DIMENSION F(6),FP(6),PRNT(5),SO(4),SN(4)
1117 INTEGER*4 N11(20)
1118 COMMON/OUT/L,TH1,TH2,H,Z,S1,DS,X,N11,J2,J5,JJ,SPACE,SIGMA
1119 *,DO,CD,AT,XXX,IOUT,ITERM
1120 COMMON/AMP/CIO,TIO,PIO,RHOO,AL,TO,CO,FR,R1,FRL,TIS
1121 COMMON/PLTE/J6,J7,J8,J9,J10,X1M,Y1M,X2M,Y2M,Y3M
1122 COMMON/ZP9/SET,XE,YE,ZE,BBE,DCE,DTE,RUE,RTE,RCE,T1E,T2E
1123 COMMON/STR95/II,M,R3
1124 COMMON/CONS/II1,II2,II3,II4,III
1125 COMMON/WGH/MERGCK,ROVO
1126 COMMON/OPT/IDEN
1127 COMMON/TPDAT/ZTRAP,DILT,PDEP,ITRAP,DILSF,ISURF,INTER
1128 LOGICAL IDEN
1129 I=III
1130 PII=3.14159
1131 IF(L)6,8,7
1132 6 L=0
1133 L19=3
1134 M1=-1
1135 M2=0
1136 Z1=-DS
1137 SNEW=PRNT(1)
1138 SN(1)=SIN(T1E)
1139 SN(2)=SIN(T2E)
1140 SN(3)=COS(T1E)
1141 SN(4)=COS(T2E)
1142 TIME=SET
1143 X=XE
1144 Y=YE
1145 Z=ZE
1146 UCC=1.0
1147 S1=PRNT(1)
1148 T3I=TIO
1149 C3I=0.
1150 IF(.NOT.IDEN)C3I=CIO
1151 IF(CIO-1.0.LT..000001)C3I=0.
1152 IF(TIO-1.0)63,64,63
1153 64 T3I=0.0
1154 63 GO TO 10
1155 8 L=1
1156 SNEW=DS+S1
1157 SO(1)=SN(1)
1158 SO(2)=SN(2)
1159 SO(3)=SN(3)
1160 SO(4)=SN(4)
1161 7 IF(M2)18,18,19
1162 19 DS=S-SNEW
1163 IF(S-S1-1.0)83,1,1
1164 18 IF(S-SNEW)20,1,1
1165 20 IF(ABS(S-SNEW)-0.001*SNEW)1,1,83
1166 1 SN(1)=SIN(F(5))
1167 SN(2)=SIN(F(6))
1168 SN(3)=COS(F(5))
1169 SN(4)=COS(F(6))
1170 X=X+0.5*(SN(3)*SN(4)+SO(3)*SO(4))*(S-S1)

```

PROGRAM UDKHDEN

```

1171      Y=Y+0.5*(SN(1)+SN(4)+SO(1)+SO(4))*(S-S1)
1172      Z1=Z
1173      Z=Z+0.5*(SN(2)+SO(2))*(S-S1)
1174      DO 9 I=1,4
1175      9 SO(I)=SN(I)
1176      10 CALL AMBIEN(Z,CI,TI,PI,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
1177      IF(F(4))21,21,55
1178      55 R14=R+SN(1)+SN(4)
1179      A=SPACE/SIGMA
1180      IF(A-.2)501,500,500
1181      501 IF(A-.95)503,503,502
1182      C **** SINGLE PLUME ****
1183      500 C1=I11
1184      C2=R14/2.
1185      C3=I12
1186      C4=2.*R14*I11
1187      C5=R14*R14/2.
1188      C6=I12
1189      C7=I11*R14
1190      GO TO 504
1191      C **** MERGING PLUMES ****
1192      502 CALL XINT(I,A,AI1,AI2,AI3)
1193      IF(MERGCK.EQ.0)WRITE(10UT,519)
1194      IF(MERGCK.EQ.0)M=M+3
1195      MERGCK=1
1196      519 FORMAT(/3X,' PLUMES MERGING'/)
1197      C1=2.*I13*AI1/PII
1198      C2=R14*AI3/PII
1199      C3=2.*I14*AI2/PII
1200      C4=4.*I13*AI1*R14/PII
1201      C5=R14*R14*AI3/PII
1202      C6=2.*I14*AI2/PII
1203      C7=2.*I13*R14*AI1/PII
1204      GO TO 504
1205      C **** MERGED PLUMES ****
1206      503 CALL XINT(I,A,AI1,AI2,AI3)
1207      C1=.45*AI3/PII
1208      C2=R14*AI3/PII
1209      C3=.31558*AI3/PII
1210      C4=2.*R14*C1
1211      C5=R14*C2
1212      C6=C3
1213      C7=0.5*C4
1214      504 IF(ABS(R14)-.001)12,12,99
1215      12 BSQR=F(1)*F(1)*C3/(F(4)*C1+C1)
1216      GO TO 15
1217      99 AA=C3*C2*C2/C1/C1-C4*C2/C1+C5
1218      BB=C4*F(1)/C1-C3*2.*C2*F(1)/C1/C1-F(4)
1219      CC=C3*F(1)*F(1)/C1/C1
1220      BSQR=BB-SQRT(BB*BB-4.*AA*CC)
1221      BSQR=BSQR/(2.*AA)
1222      15 B=SQRT(BSQR)
1223      SIGMA=B
1224      SIGMAO=SIGMA*DO
1225      DLUM=F(1)/(BSQR+C1)-C2/C1
1226      UCL=DLUM+R14
1227      DTCL=F(2)/(BSQR*(DLUM+C6+C7))
1228      DCCL=F(3)/(BSQR*(DLUM+C6+C7))
1229      C THE ABOVE CARDS DEFINE THE COEFFICIENTS OF THE CONTINUITY,MOMENTUM
1230      C AND ENERGY EQUATION FOR THE SINGLE PLUME, MERGING PLUME AND
1231      C MERGED PLUME REGIONS BASED ON PROFILES OF THE FORM (1.-(X/B)**3/2)
1232      C SQUARED
1233      DELU=(UCL-R14)/(1.0-R3)
1234      DELT=DTCL/(1.0-T3I)
1235      DELC=DCCL/(1.-C3I)

```

PROGRAM UDKHDEH

```

1236      TCL=(DTCL+TI)*TO
1237      IF(.NOT.IDEN)CCL=(DCCL+CI)*CO
1238      IF(.NOT.IDEN)CALL SIGMAT(PCL,TCL,CCL)
1239      IF(.NOT.IDEN)PCL=PCL/RHOO
1240      TCL=TCL/TO
1241      IF(IDEN)PCL=TCL
1242      IF(IDEN)CCL=DCCL
1243      IF(.NOT.IDEN)CCL=CCL/CO
1244      C DELU=(UCL-UI*S1+C2)/(UO-UI)      DELT=(TCL-TI)/(TO-TI)
1245      F55=F(5)*57.29578
1246      F66=F(6)*57.29578
1247      C CONVERT ANGLES BACK TO DEGREES
1248      UMA=0.5*(UCL+UCC)
1249      TIME=TIME+(S-S1)/UMA
1250      FRL=FR
1251      IF(FR-0.98E+6)110,112,112
1252      110 IF(ABS((PI-PCL)*B)-1.0E-6*ABS(UCL**2*FR*(PIO-1.0)))112,113,113
1253      113 FRL=FR*(PIO-1.0)*UCL=UCL/((PI-PCL)*B)
1254      112 CONTINUE
1255      C
1256      C      SOO IS THE DISTANCE ALONG THE PLUME CENTER LINE, IT IS NOT
1257      C      OUTPUT IN THIS VERSION, IT WAS IN DKHPLM (S).
1258      C      112 SOO=S/2.*DO
1259      C
1260      XO=X/2.*DO
1261      YO=Y/2.*DO
1262      ZOLD=ZNEW
1263      ZO=Z/2.*DO
1264      ZNEW=ZO
1265      TIMER=TIME*ROVO
1266      RGOLD=RQ
1267      RQ=2.*F(1)
1268      DENOLD=DEN
1269      PICL=PI-PCL
1270      PICLO=PIO-1.0
1271      DEN=PICL/PICLO
1272      XDUM=1.
1273      IF(M2)40,40,42
1274      40 IF(DSIGN(XDUM,PICL)-DSIGN(XDUM,PICLO))41,42,41
1275      41 DS=0.0
1276      M1=0
1277      M2=1
1278      ITRAP=1
1279      WRITE(IOUT,213)
1280      213 FORMAT(/4X,'PLUMES HAVE REACHED EQUILIBRIUM HEIGHT -',
1281      * ' STRATIFIED ENVIRONMENT'/)
1282      M=M+3
1283      IF(INTER.EQ.1)WRITE(ITERM,213)
1284      PRO=DENOLD/(DENOLD-DEN)
1285      DILTP=RGOLD-PRO*(RGOLD-RQ)
1286      ZP=ZOLD-PRO*(ZOLD-ZNEW)
1287      ZTRAP=PDEP-ZP
1288      42 IF(L19.LT.3)GO TO 71
1289      IF(IDEN)WRITE(IOUT,104)XO,YO,ZO,F55,F66,SIGMAO,DELU,DEN,DELC
1290      *,TIMER,RQ
1291      IF(.NOT.IDEN)WRITE(IOUT,104)XO,YO,ZO,F55,F66,SIGMAO,DEN,DELT,DELC
1292      *,TIMER,RQ
1293      104 FORMAT(1X,5F8.2,1X,F8.2,3F9.3,1X,2F9.2)
1294      M1=M1+1
1295      M=M+1
1296      L19=L19+1
1297      GO TO 43
1298      71 L19=L19+1
1299      43 IF(M1-10)36,32,32
1300      32 M1=0

```

PROGRAM UDKHDEN

```

1301      DS=2.0*DS
1302      36 IF(Z)3,3,31
1303      31 IF(M2)90,90,91
1304      91 IF(Z-21)21,90,90
1305      21 PRNT(5)=1.0
1306      WRITE(IOUT,212)
1307      212 FORMAT(/3X,' PLUMES HAVE REACHED MAXIMUM HEIGHT -' ,
1308      * ' STRATIFIED ENVIRONMENT',/)
1309      IF(J6)83,83,81
1310      90 IF(Z-H)3,3,4
1311      4 PRNT(5)=1.0
1312      WRITE(IOUT,214)
1313      214 FORMAT(/3X,' PLUME HAS REACHED WATER SURFACE',/)
1314      C IF WE HAVE REACHED THE WATER SURFACE-STOP.
1315      IF(INTER .EQ. 1)WRITE(ITERM,214)
1316      ISURF=1
1317      PROT=(ZOLD-PDEP)/(ZOLD-ZNEW)
1318      DILSF=RQOLD-PROT*(RQOLD-RQ)
1319      IF(J6)83,83,81
1320      3 SNEW=SNEW+DS
1321      S1=S
1322      UCC=UCL
1323      IF(M2)52,52,51
1324      51 DS=0.0
1325      M1=0
1326      52 IF(M-59)2,13,13
1327      13 WRITE(IOUT,207)M1
1328      207 FORMAT(1H1,/10X,20A4)
1329      IF(IDEN)WRITE(IOUT,208)
1330      IF(.NOT.IDEN)WRITE(IOUT,209)
1331      208 FORMAT(1X,5X,'X',7X,'Y',7X,'Z',6X,'TH1',5X,'TH2',6X,'WIDTH',
1332      *4X,'DUCL',4X,'DRHO',5X,'DCCL',7X,'TIME',3X,'DILUTION'/)
1333      209 FORMAT(1X,5X,'X',7X,'Y',7X,'Z',6X,'TH1',5X,'TH2',6X,'WIDTH',
1334      *4X,'DRHO',4X,'DTCL',5X,'DSCL',7X,'TIME',3X,'DILUTION'/)
1335      M=4
1336      2 IF(J6)83,83,84
1337      84 IF(SNEW-PRNT(2))80,81,81
1338      81 CONTINUE
1339      PRNT(5)=1.0
1340      RETURN
1341      80 IF(M2-1)83,85,83
1342      85 IF(ABS(Z-21)-0.05*PRNT(3))81,83,83
1343      83 RETURN
1344      END
1345      C
1346      SUBROUTINE ZFE(INLF)
1347      C THIS PROGRAM COMPUTES THE SOLUTION TO BUOYANT JET PROBLEMS WITHIN
1348      C THE ZONE OF FLOW ESTABLISHMENT, I.E. UP TO XE.
1349      C THE SOLUTION CALCULATES RU,RT,RC,B,TH1,TH2 AS FUNCTIONS OF S.
1350      IMPLICIT REAL*8(A-H,O-Z)
1351      DIMENSION PRNT(5),AUX(16,6),F(6),FP(6)
1352      INTEGER*4 N11(20)
1353      COMMON/AMB/CIO,TIO,PIO,RHOO,AL,TO,CO,FR,R1,FRL,TIS
1354      COMMON/OUT/L,TH1,TH2,H,Z,SNEW,DS,K,N11,J2,J5,JJ,SPACE,SIGMA
1355      * ,DO,CD,A7,XXX,IOUT,ITERM
1356      COMMON/ZF9/SET,XE,YE,ZE,DDE,DCE,DTE,RUE,RTE,RCE,TIE,T2E
1357      EXTERNAL DERIV1,OUTP1
1358      CALL AMBIEN(Z,CIO,TIO,PIO,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
1359      TH1=0.01745329*TH1
1360      TH2=0.01745329*TH2
1361      PRNT(1)=0.0
1362      PRNT(2)=100.0
1363      PRNT(3)=0.1
1364      PRNT(4)=0.001
1365      IF(ABS(COS(TH2))-0.01)1,1,2

```

PROGRAM UDKHDEN

```

1366      1 TH1=TH2
1367      2 N=6
1368      L=0
1369      DO 3 I=1,6
1370      3 FP(I)=0.16666667
1371      F(1)=1.0
1372      F(2)=1.0
1373      F(3)=1.0
1374      F(4)=0.0
1375      F(5)=TH1
1376      F(6)=TH2
1377      C F(1)=RU, F(2)=RT, F(3)=RC, F(4)=B, F(5)=TH1, F(6)=TH2
1378      IHLF=0
1379      CALL OUTP1(PRMT(1),F,FP,IHLF,N,PRMT)
1380      CALL HPCG(PRMT,F,FP,N,IHLF,DERIV1,OUTP1,AUX)
1381      IF(IHLF .GE. 11)GO TO 95
1382      F(1)=RUE
1383      IF(F(1) .LE. 0.)F(1)=1.E-03
1384      CALL AMBIEN(Z,CI,TI,PI,TIZ,CIZ,PIZ,VTI,VTI,VUI,VVI,R)
1385      F(2)=1.-TI
1386      F(3)=1.-CI
1387      F(4)=BBE
1388      F(5)=T1E=0.01745329
1389      F(6)=T2E=0.01745329
1390      TH1=F(5)
1391      TH2=F(6)
1392      DO 9 I=1,6
1393      9 FP(I)=0.16666667
1394      PRMT(1)=SET
1395      PRMT(2)=SET+100.0
1396      PRMT(3)=J.1
1397      PRMT(4)=0.001
1398      IHLF=0
1399      CALL OUTP1(PRMT(1),F,FP,IHLF,N,PRMT)
1400      CALL HPCG(PRMT,F,FP,N,IHLF,DERIV1,OUTP1,AUX)
1401      95 RETURN
1402      END
1403      C
1404      SUBROUTINE DERIV1(S,F,FP)
1405      IMPLICIT REAL*8(A-H,O-Z)
1406      DIMENSION F(6),FP(6),A(4,4),A2(3,3),A3(3,3)
1407      INTEGER*4 N11(20)
1408      COMMON/AMB/CIO,TIO,PIO,RHOO,AL,TO,CO,FR,R1,FRL,TIS
1409      COMMON/OUT/L,TH1,TH2,H,Z,SNEW,DS,K,N11,J2,J5,JJ,SPACE,SIGMA
1410      *,DO,CD,A7,XXX,IOUT,ITERM
1411      COMMON/ZF9/SET,XE,YE,ZE,BBE,DCE,DTE,RUE,RTE,RCE,T1E,T2E
1412      C
1413      E1=1.06
1414      E2=34.0
1415      E3=6.0
1416      E4=.4
1417      S1=SIN(F(5))
1418      S2=SIN(F(6))
1419      C1=COS(F(5))
1420      C2=COS(F(6))
1421      S12=S1*C2
1422      CALL AMBIEN(Z,CI,TI,PI,TIZ,CIZ,PIZ,VTE,VCE,VUE,VVE,R)
1423      R12=R*S1*C2
1424      D10=1.-TI
1425      D10=1.-CI
1426      E=E1*(0.0204+0.0144*F(4))*(1.0+E2*S2/FR)*(ABS(1.0-R12)+E3*R
1427      **SQRT(1.0-S12**2))*(1-E4/SPACE)
1428      C E IS THE ENTRAINMENT FUNCTION.
1429      D1=.45+.55*R12
1430      D11=2.*(1.2857+.37143*R12)

```

PROGRAM UDKHDEH

```

1431      D2=.31558+.13442*R12
1432      D3=2.*(.06676+.06181*R12)
1433      D4=.31558+.26884*R12+.41558*R12*R12
1434      D5=2.*(.06676+.12362*R12+.30962*R12*R12)
1435      D6=(PI-1.0)*(F(2)*F(2)+0.5+0.12857*F(4)*F(4)+.45*F(2)*F(4))
1436      Q=0.5*F(1)*F(1)+F(1)*F(4)*D4+F(4)*F(4)*0.5*D5-0.25*E*E
1437      AAI=0.5*F(1)*F(1)+D1*F(1)*F(4)+D11*0.5*F(4)*F(4)
1438      A(1,1)=F(1)+D1*F(4)
1439      A(1,2)=0.0
1440      A(1,3)=0.0
1441      A(1,4)=D1*F(1)+D11*F(4)
1442      C FIRST EQUATION IS CONTINUITY.
1443      A(2,1)=0.0
1444      A(2,2)=D10*F(2)+D2*D10*F(4)
1445      A(2,3)=0.0
1446      A(2,4)=D10*(D2*F(2)+D3*F(4))
1447      C SECOND EQUATION IS ENERGY.
1448      A(3,1)=0.0
1449      A(3,2)=0.0
1450      A(3,3)=D10*(F(3)+D2*F(4))
1451      A(3,4)=D10*(D2*F(3)+D3*F(4))
1452      C THIRD EQUATION IS CONCENTRATION.
1453      A(4,1)=F(1)+D4*F(4)
1454      A(4,2)=0.0
1455      A(4,3)=0.0
1456      A(4,4)=D4*F(1)+D5*F(4)
1457      C FOURTH EQUATION IS MOMENTUM
1458      FP(1)=E
1459      TIZZ=T1Z
1460      FP(2)=TIZZ*S2*(-AAI+0.5*F(2)+2*F(2)*F(4)+D2+0.5*F(4)+2*D3)
1461      FP(2)=FP(2)+VTE
1462      FP(3)=C1Z*S2*(-AAI+0.5*F(3)+2*F(3)*F(4)+D2+0.5*F(4)+2*D3)
1463      FP(3)=FP(3)+VCE
1464      IF(L-2)3,4,3
1465      4 A(2,2)=0.5*D3*F(4)+2
1466      A(2,4)=D3*F(2)*F(4)
1467      A(3,3)=A(2,2)
1468      A(3,4)=D3*F(3)*F(4)
1469      FP(2)=-TIZZ*S2*AAI+VTE*DTE
1470      FP(3)=-C1Z*S2*AAI+VCE*DCE
1471      D6=(PI-1.0)*0.12857*F(4)+2
1472      3 IF(FR-0.98E+6)11,12,12
1473      11 F8=D6/(FR*(PI0-1.0))
1474      GO TO 13
1475      12 F8=0.0
1476      13 FP(4)=F8*S2+E*R12+VUE
1477      J=4
1478      IF(DC0.EQ.0.0.AND.D10.EQ.0.0)GO TO 10
1479      IF(D10.EQ.0.0)GO TO 6
1480      IF(DC0.EQ.0.0)GO TO 20
1481      CALL SIMQ(A,FP,J,K)
1482      GO TO 7
1483      10 J=2
1484      A2(1,1)=A(1,1)
1485      A2(1,2)=A(1,4)
1486      A2(2,1)=A(4,1)
1487      A2(2,2)=A(4,4)
1488      FP(2)=FP(4)
1489      CALL SIMQ(A2,FP,J,K)
1490      GO TO 7
1491      20 J=3
1492      A3(1,1)=A(1,1)
1493      A3(1,2)=0.0
1494      A3(1,3)=A(1,4)
1495      A3(2,1)=0.0

```

PROGRAM UDKHDEN

```

1496      A3(2,2)=A(2,2)
1497      A3(2,3)=A(2,4)
1498      A3(3,1)=A(4,1)
1499      A3(3,2)=0.
1500      A3(3,3)=A(4,4)
1501      FP(3)=FP(4)
1502      CALL SIMQ(A3,FP,J,K)
1503      GO TO 7
1504      6 J=3
1505      A3(1,1)=A(1,1)
1506      A3(1,2)=0.
1507      A3(1,3)=A(1,4)
1508      A3(2,1)=0.
1509      A3(2,2)=A(3,3)
1510      A3(2,3)=A(3,4)
1511      A3(3,1)=A(4,1)
1512      A3(3,2)=0.
1513      A3(3,3)=A(4,4)
1514      FP(2)=FP(3)
1515      FP(3)=FP(4)
1516      CALL SIMQ(A3,FP,J,K)
1517      7 IF(K)1,2,1
1518      2 IF(J-3)131,132,136
1519      131 FP(4)=FP(2)
1520      FP(3)=0.
1521      FP(2)=0.
1522      GO TO 136
1523      132 IF(DCO.NE.O.)GO TO 134
1524      FP(4)=FP(3)
1525      FP(3)=0.
1526      GO TO 136
1527      134 FP(4)=FP(3)
1528      FP(3)=FP(2)
1529      FP(2)=0.
1530      136 UNX=-(C2**2+C1*S1)
1531      UNY=S2**2+C2**2+C1**2
1532      UNZ=-(S2*C2*S1)
1533      UN=SQRT(UNX**2+UNY**2+UNZ**2)
1534      CDP=CD/SPACE
1535      FD=CDP*R*R*UN/6.2832
1536      FDY=0.
1537      YY=ABS(C1*C2)
1538      IF(YY.GT.O.)FDY=FD*(S2**2+C2**2+C1**2-S1**2+S2**2)/(Q*C1*C2)
1539      FP(5)=E*R*C1/(Q*C2)+FDY
1540      FP(6)=(F8*C2-(E*R+FD)*S1*S2)/Q
1541      RETURN
1542      1 WRITE(IOUT,100)
1543      100 FORMAT(1X,' SINGULAR MATRIX-EXECUTION TERMINATED AT THIS POINT')
1544      CALL EXIT
1545      END
1546      C
1547      SUBROUTINE OUTP1(S,F,FP,IN,N,PRMT)
1548      IMPLICIT REAL*8(A-H,O-Z)
1549      COMMON/AMB/CIO,TIO,PIO,RHOO,AL,TO,CO,FR,R1,FRL,TIS
1550      COMMON/OUT/L,TH1,TH2,H,Z,SNEW,DS,K,N11,J2,J5,JJ,SPACE,SIGMA
1551      *,DO,CD,A7,XXX,IOUT,ITERM
1552      COMMON/ZF9/SET,XE,YE,ZE,BBE,DCE,DTE,RUE,RTE,RCE,T1E,T2E
1553      COMMON/STR95/II,M,R3
1554      COMMON/WGM/MERGCK,ROVO
1555      DIMENSION F(6),FP(6),PRMT(5),SO(4),SN(4)
1556      INTEGER*4 N11(20)
1557      IF(L)1,1,2
1558      1 SN(1)=SIN(TH1)
1559      SN(2)=SIN(TH2)
1560      SN(3)=COS(TH1)

```

PROGRAM UDKHDEM

```

1561      SN(4)=COS(TH2)
1562      TJ=PRNT(1)
1563      DS=PRNT(3)
1564      RUN=0.0
1565      RTN=0.0
1566      RCN=0.0
1567      BN=1.0
1568      SSN=0.0
1569      TIN=TH1*57.29578
1570      T2N=TH2*57.29578
1571      XN=0.0
1572      YN=0.0
1573      ZN=0.0
1574      X=0.0
1575      Y=0.0
1576      Z=0.0
1577      C      JP=100
1578      IF(L) 30,31,30
1579      31 L=1
1580      GO TO 6
1581      30 L=2
1582      X=XE
1583      Y=YE
1584      Z=ZE
1585      GO TO 6
1586      2 IF(ABS(S-TJ)-0.01) 3,3,4
1587      3 SN(1)=SIN(F(5))
1588      SN(2)=SIN(F(6))
1589      SN(3)=COS(F(5))
1590      SN(4)=COS(F(6))
1591      X=X+0.5*(SN(3)*SN(4)+SO(3)*SO(4))*DS
1592      Y=Y+0.5*(SN(1)*SN(4)+SO(1)*SO(4))*DS
1593      Z=Z+0.5*(SN(2)+SO(2))*DS
1594      6 DO 5 I=1,4
1595      5 SO(I)=SN(I)
1596      TIME=S
1597      CALL AMBIEN(Z,C1,TI,PI,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
1598      F55=F(5)*57.29578
1599      F66=F(6)*57.29578
1600      C
1601      C F(2)=RT OR (TH-TI)/TO, F(3) SAME FOR C
1602      C
1603      RUO=RUN
1604      RTO=RTN
1605      RCO=RCN
1606      DO=BN
1607      SS=SSN
1608      T1O=TIN
1609      T2O=T2N
1610      XO=XN
1611      YO=YN
1612      ZO=ZN
1613      RUN=F(1)
1614      RTN=F(2)
1615      RCN=F(3)
1616      BN=F(4)
1617      SSN=S
1618      TIN=F55
1619      T2N=F66
1620      XN=X
1621      YN=Y
1622      ZN=Z
1623      IF(L-2) 20,21,20
1624      20 IF(F(2)) 14,14,13
1625      14 CONTINUE

```

PROGRAM UKHDEH

```

1561      SN(4)=COS(TM2)
1562      TJ=PRNT(1)
1563      DS=PRNT(3)
1564      RUN=0.0
1565      RTN=0.0
1566      RCN=0.0
1567      BN=1.0
1568      SSN=0.0
1569      TIN=TM1+57.29578
1570      T2N=TM2+57.29578
1571      XN=0.0
1572      YN=0.0
1573      ZN=0.0
1574      X=0.0
1575      Y=0.0
1576      Z=0.0
1577      C      JP=100
1578      IF(L)30,31,30
1579      31 L=1
1580      GO TO 6
1581      30 L=2
1582      X=XE
1583      Y=YE
1584      Z=ZE
1585      GO TO 6
1586      2 IF(ABS(S-TJ)-0.01)3,3,4
1587      3 SN(1)=SIN(F(5))
1588      SN(2)=SIN(F(6))
1589      SN(3)=COS(F(5))
1590      SN(4)=COS(F(6))
1591      X=X+0.5*(SN(3)*SN(4)+S0(3)*S0(4))*DS
1592      Y=Y+0.5*(SN(1)*SN(4)+S0(1)*S0(4))*DS
1593      Z=Z+0.5*(SN(2)+S0(2))*DS
1594      6 DO 5 I=1,4
1595      5 S0(I)=SN(I)
1596      TIME=S
1597      CALL AMBIEN(Z,CI,TI,PI,TIZ,CIZ,PIZ,VTI,VC1,VUI,VVI,R)
1598      F55=F(5)+57.29578
1599      F66=F(6)+57.29578
1600      C
1601      C F(2)=RT OR (TM-TI)/TO, F(3) SAME FOR C
1602      C
1603      RUO=RUN
1604      RTO=RTN
1605      RCO=RCN
1606      BO=BN
1607      SS=SSN
1608      TIO=TIN
1609      TZO=T2N
1610      XO=XN
1611      YO=YN
1612      ZO=ZN
1613      RUN=F(1)
1614      RTN=F(2)
1615      RCN=F(3)
1616      BN=F(4)
1617      SSN=S
1618      TIN=F55
1619      T2N=F66
1620      XN=X
1621      YN=Y
1622      ZN=Z
1623      IF(L-2)20,21,20
1624      20 IF(F(2))14,14,13
1625      14 CONTINUE

```

PROGRAM UDKHDEN

```

1626      SET=SSN-RTN*(SSN-SS)/(RTN-RTO)
1627      RUE=RUE*(SET-SS)*(RUE-RUN)/(SS-SSN)
1628      RTE=RTO*(SET-SS)*(RTO-RTN)/(SS-SSN)
1629      RCE=RCO*(SET-SS)*(RCO-RCN)/(SS-SSN)
1630      BBE=BO*(SET-SS)*(BO-BN)/(SS-SSN)
1631      T1E=T1O*(SET-SS)*(T1O-T1N)/(SS-SSN)
1632      T2E=T2O*(SET-SS)*(T2O-T2N)/(SS-SSN)
1633      XE=XO*(SET-SS)*(XO-XN)/(SS-SSN)
1634      YE=YO*(SET-SS)*(YO-YN)/(SS-SSN)
1635      ZE=ZO*(SET-SS)*(ZO-ZN)/(SS-SSN)
1636      S00=SET/2.*D0
1637      WRITE(IOUT,200)S00
1638      200 FORMAT(1H+,57X,'STARTING LENGTH=',P9.3)
1639      L=-2
1640      PRMT(5)=1.0
1641      RETURN
1642      13 TJ=TJ+PRMT(3)
1643      RETURN
1644      40 TJ=TJ+PRMT(3)
1645      RETURN
1646      21 IF(F(1))10,10,40
1647      10 PRMT(5)=1.0
1648      SET=SSN-RUN*(SSN-SS)/(RUN-RUE)
1649      RUE=RUE*(SET-SS)*(RUE-RUN)/(SS-SSN)
1650      BBE=BO*(SET-SS)*(BO-BN)/(SS-SSN)
1651      RTE=RTO*(SET-SS)*(RTO-RTN)/(SS-SSN)
1652      RCE=RCO*(SET-SS)*(RCO-RCN)/(SS-SSN)
1653      T1E=T1O*(SET-SS)*(T1O-T1N)/(SS-SSN)
1654      T2E=T2O*(SET-SS)*(T2O-T2N)/(SS-SSN)
1655      XE=XO*(SET-SS)*(XO-XN)/(SS-SSN)
1656      YE=YO*(SET-SS)*(YO-YN)/(SS-SSN)
1657      ZE=ZO*(SET-SS)*(ZO-ZN)/(SS-SSN)
1658      CALL AMBIEN(ZE,CI,TI,PI,TIZ,CIZ,PIZ,VTI,VCI,VUI,VVI,R)
1659      4 RETURN
1660      END
1661      C
1662      SUBROUTINE SIMQ(A,B,N,KS)
1663      C SIMQ SOLVES THE SYSTEM OF J LINEAR EQUATIONS FOR THE FP. A IS THE
1664      C COEFFICIENT MATRIX IN EQN A.FP=FP. THE INPUT FP ARE DESTROYED
1665      C BY SIMQ AND REPLACED BY THE SOLUTION VECTOR, FP.
1666      IMPLICIT REAL*8(A-H,O-Z)
1667      DIMENSION A(16),B(6)
1668      C
1669      C FORWARD SOLUTION
1670      C
1671      TOL=0.0
1672      KS=0
1673      JJ=-N
1674      DO 65 J=1,N
1675      JY=J+1
1676      JJ=JJ+N+1
1677      BIGA=0
1678      IT=JJ-J
1679      DO 30 I=J,N
1680      C
1681      C SEARCH FOR MAXIMUM COEFFICIENT IN COLUMN
1682      C
1683      IJ=IT+I
1684      IF(ABS(BIGA)-ABS(A(IJ)))20,30,30
1685      20 BIGA=A(IJ)
1686      IMAX=I
1687      30 CONTINUE
1688      C
1689      C TEST FOR PIVOT LESS THAN TOLERANCE (SINGULAR MATRIX)
1690      C

```

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

SIMQ

PROGRAM UBKHEN

```

1691      IF (ABS(BIGA)-TOL)35,35,40      SIMQ
1692 35 KS=1      SIMQ
1693      RETURN      SIMQ
1694 C      SIMQ
1695 C      INTERCHANGE ROWS IF NECESSARY      SIMQ
1696 C      SIMQ
1697 40 I1=J+N*(J-2)      SIMQ
1698      IT=IMAX-J      SIMQ
1699      DO 50 K=J,N      SIMQ
1700      I1=I1+N      SIMQ
1701      I2=I1+IT      SIMQ
1702      SAVE=A(I1)      SIMQ
1703      A(I1)=A(I2)      SIMQ
1704      A(I2)=SAVE      SIMQ
1705 C      SIMQ
1706 C      DIVIDE EQUATION BY LEADING COEFFICIENT      SIMQ
1707 C      SIMQ
1708 50 A(I1)=A(I1)/BIGA      SIMQ
1709      SAVE=B(IMAX)      SIMQ
1710      B(IMAX)=B(J)      SIMQ
1711      B(J)=SAVE/BIGA      SIMQ
1712 C      SIMQ
1713 C      ELIMINATE NEXT VARIABLE      SIMQ
1714 C      SIMQ
1715      IF (J=N)55,70,55      SIMQ
1716 55 IQS=N*(J-1)      SIMQ
1717      DO 65 IX=JY,N      SIMQ
1718      IXJ=IQS+IX      SIMQ
1719      IT=J-IX      SIMQ
1720      DO 60 JX=JY,N      SIMQ
1721      IXJX=N*(JX-1)+IX      SIMQ
1722      JJX=IXJX+IT      SIMQ
1723 60 A(IXJX)=A(IXJX)-(A(IXJ)*A(JJX))      SIMQ
1724 65 B(IX)=B(IX)-(B(J)*A(IXJ))      SIMQ
1725 C      SIMQ
1726 C      BACK SOLUTION      SIMQ
1727 C      SIMQ
1728 70 NY=N-1      SIMQ
1729      IT=N+N      SIMQ
1730      DO 80 J=1,NY      SIMQ
1731      IA=IT-J      SIMQ
1732      IB=N-J      SIMQ
1733      IC=N      SIMQ
1734      DO 80 K=1,J      SIMQ
1735      B(IB)=B(IB)-A(IA)*B(IC)      SIMQ
1736      IA=IA-N      SIMQ
1737 80 IC=IC-1      SIMQ
1738      RETURN      SIMQ
1739      END      SIMQ
1740 C      HPCG
1741 C      .....HPCG
1742      SUBROUTINE HPCG(PRMT,Y,DERV,NDIM,IHLF,FCT,OUTP,AUX)      HPCG
1743 C      SUBROUTINE HPCG      HPCG
1744 C      HPCG
1745 C      PURPOSE      HPCG
1746 C      TO SOLVE A SYSTEM OF FIRST ORDER ORDINARY GENERAL      HPCG
1747 C      DIFFERENTIAL EQUATIONS WITH GIVEN INITIAL VALUES.      HPCG
1748 C      HPCG
1749 C      USAGE      HPCG
1750 C      CALL HPCG (PRMT,Y,DERV,NDIM,IHLF,FCT,OUTP,AUX)      HPCG
1751 C      PARAMETERS FCT AND OUTP REQUIRE AN EXTERNAL STATEMENT.      HPCG
1752 C      HPCG
1753 C      DESCRIPTION OF PARAMETERS      HPCG
1754 C      PRMT - AN INPUT AND OUTPUT VECTOR WITH DIMENSION GREATER      HPCG
1755 C      OR EQUAL TO 5, WHICH SPECIFIES THE PARAMETERS OF      HPCG

```

PROGRAM UDKNDEN

```

1756 C      THE INTERVAL AND OF ACCURACY AND WHICH SERVES FOR HPCG
1757 C      COMMUNICATION BETWEEN OUTPUT SUBROUTINE (FURNISHED HPCG
1758 C      BY THE USER) AND SUBROUTINE HPCG. EXCEPT PRMT(5) HPCG
1759 C      THE COMPONENTS ARE NOT DESTROYED BY SUBROUTINE HPCG
1760 C      HPCG AND THEY ARE HPCG
1761 C      PRMT(1)- LOWER BOUND OF THE INTERVAL (INPUT), HPCG
1762 C      PRMT(2)- UPPER BOUND OF THE INTERVAL (INPUT), HPCG
1763 C      PRMT(3)- INITIAL INCREMENT OF THE INDEPENDENT VARIABLE HPCG
1764 C      (INPUT), HPCG
1765 C      PRMT(4)- UPPER ERROR BOUND (INPUT). IF ABSOLUTE ERROR IS HPCG
1766 C      GREATER THAN PRMT(4), INCREMENT GETS HALVED. HPCG
1767 C      IF INCREMENT IS LESS THAN PRMT(3) AND ABSOLUTE HPCG
1768 C      ERROR LESS THAN PRMT(4)/50, INCREMENT GETS DOUBLED. HPCG
1769 C      THE USER MAY CHANGE PRMT(4) BY MEANS OF HIS HPCG
1770 C      OUTPUT SUBROUTINE. HPCG
1771 C      PRMT(5)- NO INPUT PARAMETER. SUBROUTINE HPCG INITIALIZES HPCG
1772 C      PRMT(5)=0. IF THE USER WANTS TO TERMINATE HPCG
1773 C      SUBROUTINE HPCG AT ANY OUTPUT POINT, HE HAS TO HPCG
1774 C      CHANGE PRMT(5) TO NON-ZERO BY MEANS OF SUBROUTINE HPCG
1775 C      OUTP. FURTHER COMPONENTS OF VECTOR PRMT ARE HPCG
1776 C      FEASIBLE IF ITS DIMENSION IS DEFINED GREATER HPCG
1777 C      THAN 5. HOWEVER SUBROUTINE HPCG DOES NOT REQUIRE HPCG
1778 C      AND CHANGE THEM. NEVERTHELESS THEY MAY BE USEFUL HPCG
1779 C      FOR HANDING RESULT VALUES TO THE MAIN PROGRAM HPCG
1780 C      (CALLING HPCG) WHICH ARE OBTAINED BY SPECIAL HPCG
1781 C      MANIPULATIONS WITH OUTPUT DATA IN SUBROUTINE OUTP. HPCG
1782 C      Y - INPUT VECTOR OF INITIAL VALUES. (DESTROYED) HPCG
1783 C      LATER ON Y IS THE RESULTING VECTOR OF DEPENDENT HPCG
1784 C      VARIABLES COMPUTED AT INTERMEDIATE POINTS X. HPCG
1785 C      DERY - INPUT VECTOR OF ERROR WEIGHTS. (DESTROYED) HPCG
1786 C      THE SUM OF ITS COMPONENTS MUST BE EQUAL TO 1. HPCG
1787 C      LATER ON DERY IS THE VECTOR OF DERIVATIVES, WHICH HPCG
1788 C      BELONG TO FUNCTION VALUES Y AT A POINT X. HPCG
1789 C      NDIM - AN INPUT VALUE, WHICH SPECIFIES THE NUMBER OF HPCG
1790 C      EQUATIONS IN THE SYSTEM. HPCG
1791 C      INLF - AN OUTPUT VALUE, WHICH SPECIFIES THE NUMBER OF HPCG
1792 C      BISECTIONS OF THE INITIAL INCREMENT. IF INLF GETS HPCG
1793 C      GREATER THAN 10, SUBROUTINE HPCG RETURNS WITH HPCG
1794 C      ERROR MESSAGE INLF=11 INTO MAIN PROGRAM. HPCG
1795 C      ERROR MESSAGE INLF=12 OR INLF=13 APPEARS IN CASE HPCG
1796 C      PRMT(3)=0 OR IN CASE SIGN(PRMT(3)).NE.SIGN(PRMT(2)-HPCG
1797 C      PRMT(1)) RESPECTIVELY. HPCG
1798 C      FCT - THE NAME OF AN EXTERNAL SUBROUTINE USED. IT HPCG
1799 C      COMPUTES THE RIGHT HAND SIDES DERY OF THE SYSTEM HPCG
1800 C      TO GIVEN VALUES OF X AND Y. ITS PARAMETER LIST HPCG
1801 C      MUST BE X,Y,DERY. THE SUBROUTINE SHOULD NOT HPCG
1802 C      DESTROY X AND Y. HPCG
1803 C      OUTP - THE NAME OF AN EXTERNAL OUTPUT SUBROUTINE USED. HPCG
1804 C      ITS PARAMETER LIST MUST BE X,Y,DERY,INLF,NDIM,PRMT. HPCG
1805 C      NONE OF THESE PARAMETERS (EXCEPT, IF NECESSARY, HPCG
1806 C      PRMT(4),PRMT(5),...) SHOULD BE CHANGED BY HPCG
1807 C      SUBROUTINE OUTP. IF PRMT(5) IS CHANGED TO NON-ZERO, HPCG
1808 C      SUBROUTINE HPCG IS TERMINATED. HPCG
1809 C      AUX - AN AUXILIARY STORAGE ARRAY WITH 16 ROWS AND NDIM HPCG
1810 C      COLUMNS. HPCG
1811 C      HPCG
1812 C      REMARKS HPCG
1813 C      THE PROCEDURE TERMINATES AND RETURNS TO CALLING PROGRAM, IF HPCG
1814 C      (1) MORE THAN 10 BISECTIONS OF THE INITIAL INCREMENT ARE HPCG
1815 C      NECESSARY TO GET SATISFACTORY ACCURACY (ERROR MESSAGE HPCG
1816 C      INLF=11), HPCG
1817 C      (2) INITIAL INCREMENT IS EQUAL TO 0 OR HAS WRONG SIGN HPCG
1818 C      (ERROR MESSAGES INLF=12 OR INLF=13), HPCG
1819 C      (3) THE WHOLE INTEGRATION INTERVAL IS WORKED THROUGH, HPCG
1820 C      (4) SUBROUTINE OUTP HAS CHANGED PRMT(5) TO NON-ZERO. HPCG

```

PROGRAM UMKNEN

```

1821 C
1822 C SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED HPCG
1823 C THE EXTERNAL SUBROUTINES FCT(X,Y,DERY) AND HPCG
1824 C OUTP(X,Y,DERY,INLF,NDIM,PRNT) MUST BE FURNISHED BY THE USER. HPCG
1825 C HPCG
1826 C METHOD HPCG
1827 C EVALUATION IS DONE BY MEANS OF HARRINGS MODIFIED PREDICTOR- HPCG
1828 C CORRECTOR METHOD. IT IS A FOURTH ORDER METHOD, USING 4 HPCG
1829 C PRECEDING POINTS FOR COMPUTATION OF A NEW VECTOR Y OF THE HPCG
1830 C DEPENDENT VARIABLES. HPCG
1831 C FOURTH ORDER RUNGE-KUTTA METHOD SUGGESTED BY RALSTON IS HPCG
1832 C USED FOR ADJUSTMENT OF THE INITIAL INCREMENT AND FOR HPCG
1833 C COMPUTATION OF STARTING VALUES. HPCG
1834 C SUBROUTINE HPCG AUTOMATICALLY ADJUSTS THE INCREMENT DURING HPCG
1835 C THE WHOLEa COMPUTATION BY HALVING OR DOUBLING. HPCG
1836 C TO GET FULL FLEXIBILITY IN OUTPUT, AN OUTPUT SUBROUTINE HPCG
1837 C MUST BE CODED BY THE USER. HPCG
1838 C FOR REFERENCE, SEE HPCG
1839 C (1) RALSTON/WILF, MATHEMATICAL METHODS FOR DIGITAL HPCG
1840 C COMPUTERS, WILEY, NEW YORK/LONDON, 1960, PP.95-109. HPCG
1841 C (2) RALSTON, RUNGE-KUTTA METHODS WITH MINIMUM ERROR BOUNDS, HPCG
1842 C NTAC, VOL.16, ISS.80 (1962), PP.431-437. HPCG
1843 C HPCG
1844 C ..... HPCG
1845 C HPCG
1846 C HPCG
1847 C HPCG
1848 C IMPLICIT REAL*8(A-H,O-Z)
1849 C DIMENSION PRNT(5),Y(6),DERY(6),AUX(16,6) HPCG
1850 C HPCG
1851 C HPCG
1852 C N=1 HPCG
1853 C INLF=0 HPCG
1854 C X=PRNT(1) HPCG
1855 C H=PRNT(3) HPCG
1856 C PRNT(5)=0. HPCG
1857 C DO 1 I=1,NDIM HPCG
1858 C AUX(16,I)=0. HPCG
1859 C AUX(15,I)=DERY(I) HPCG
1860 C 1 AUX(1,I)=Y(I) HPCG
1861 C IF(N=(PRNT(2)-X))3,2,4 HPCG
1862 C HPCG
1863 C ERROR RETURNS HPCG
1864 C 2 INLF=12 HPCG
1865 C GOTO 4 HPCG
1866 C 3 INLF=13 HPCG
1867 C HPCG
1868 C COMPUTATION OF DERY FOR STARTING VALUES HPCG
1869 C 4 CALL FCT(X,Y,DERY) HPCG
1870 C HPCG
1871 C RECORDING OF STARTING VALUES HPCG
1872 C CALL OUTP(X,Y,DERY,INLF,NDIM,PRNT) HPCG
1873 C IF(PRNT(5))6,5,6 HPCG
1874 C 5 IF(INLF)7,7,6 HPCG
1875 C 6 RETURN HPCG
1876 C 7 DO 8 I=1,NDIM HPCG
1877 C 8 AUX(8,I)=DERY(I) HPCG
1878 C HPCG
1879 C COMPUTATION OF AUX(2,I) HPCG
1880 C ISW=1 HPCG
1881 C GOTO 100 HPCG
1882 C HPCG
1883 C 9 X=X+H HPCG
1884 C DO 10 I=1,NDIM HPCG
1885 C 10 AUX(2,I)=Y(I) HPCG

```

PROGRAM UBKHODN

1886	C		HPCG
1887	C	INCREMENT H IS TESTED BY MEANS OF BISECTION	HPCG
1888	11	IHLF=IHLF+1	HPCG
1889		X=X-H	HPCG
1890		DO 12 I=1,NDIM	HPCG
1891	12	AUX(4,I)=AUX(2,I)	HPCG
1892		H=.5*H	HPCG
1893		N=1	HPCG
1894		ISW=2	HPCG
1895		GOTO 100	HPCG
1896	C		HPCG
1897	13	X=X+H	HPCG
1898		CALL FCT(X,Y,DERY)	HPCG
1899		N=2	HPCG
1900		DO 14 I=1,NDIM	HPCG
1901		AUX(2,I)=Y(I)	HPCG
1902	14	AUX(9,I)=DERY(I)	HPCG
1903		ISW=3	HPCG
1904		GOTO 100	HPCG
1905	C		HPCG
1906	C	COMPUTATION OF TEST VALUE DELT	HPCG
1907	15	DELT=0.	HPCG
1908		DO 16 I=1,NDIM	HPCG
1909	16	DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))	HPCG
1910		DELT=.06666667*DELT	HPCG
1911		IF(DELT-PRMT(4))19,19,17	HPCG
1912	17	IF(IHLF-10)11,18,18	HPCG
1913	C		HPCG
1914	C	NO SATISFACTORY ACCURACY AFTER 10 BISECTIONS. ERROR MESSAGE.	HPCG
1915	18	IHLF=11	HPCG
1916		X=X+H	HPCG
1917		GOTO 4	HPCG
1918	C		HPCG
1919	C	THERE IS SATISFACTORY ACCURACY AFTER LESS THAN 11 BISECTIONS.	HPCG
1920	19	X=X+H	HPCG
1921		CALL FCT(X,Y,DERY)	HPCG
1922		DO 20 I=1,NDIM	HPCG
1923		AUX(3,I)=Y(I)	HPCG
1924	20	AUX(10,I)=DERY(I)	HPCG
1925		N=3	HPCG
1926		ISW=4	HPCG
1927		GOTO 100	HPCG
1928	C		HPCG
1929	21	N=1	HPCG
1930		X=X+H	HPCG
1931		CALL FCT(X,Y,DERY)	HPCG
1932		X=PRMT(1)	HPCG
1933		DO 22 I=1,NDIM	HPCG
1934		AUX(11,I)=DERY(I)	HPCG
1935	22	Y(I)=AUX(1,I)+H*(.375*AUX(8,I)+.7916667*AUX(9,I) - .2083333*AUX(10,I)+.04166667*DERY(I))	HPCG
1936			HPCG
1937	23	X=X+H	HPCG
1938		N=N+1	HPCG
1939		CALL FCT(X,Y,DERY)	HPCG
1940		CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)	HPCG
1941		IF(PRMT(5))6,24,6	HPCG
1942	24	IF(N-4)25,200,200	HPCG
1943	25	DO 26 I=1,NDIM	HPCG
1944		AUX(N,I)=Y(I)	HPCG
1945	26	AUX(N+7,I)=DERY(I)	HPCG
1946		IF(N-3)27,29,200	HPCG
1947	C		HPCG
1948	27	DO 28 I=1,NDIM	HPCG
1949		DELT=AUX(9,I)+AUX(9,I)	HPCG
1950		DELT=DELT+DELT	HPCG

PROGRAM UBXHNDEN

```

1951      28 Y(I)=AUX(1,I)+.3333333*H*(AUX(8,I)+DELTA+AUX(10,I))      HPCG
1952      GOTO 23      HPCG
1953      C      HPCG
1954      29 DO 30 I=1,NDIM      HPCG
1955      DELTA=AUX(9,I)+AUX(10,I)      HPCG
1956      DELTA=DELTA+DELTA+DELTA      HPCG
1957      30 Y(I)=AUX(1,I)+.375*H*(AUX(8,I)+DELTA+AUX(11,I))      HPCG
1958      GOTO 23      HPCG
1959      C      HPCG
1960      C      THE FOLLOWING PART OF SUBROUTINE HPCG COMPUTES BY MEANS OF      HPCG
1961      C      RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING      HPCG
1962      C      PREDICTOR-CORRECTOR METHOD.      HPCG
1963      100 DO 101 I=1,NDIM      HPCG
1964      Z=H*AUX(N+7,I)      HPCG
1965      AUX(5,I)=Z      HPCG
1966      101 Y(I)=AUX(N,I)+.4*Z      HPCG
1967      C      Z IS AN AUXILIARY STORAGE LOCATION      HPCG
1968      C      HPCG
1969      Z=X+.4*H      HPCG
1970      CALL FCT(Z,Y,DERY)      HPCG
1971      DO 102 I=1,NDIM      HPCG
1972      Z=H*DERY(I)      HPCG
1973      AUX(6,I)=Z      HPCG
1974      102 Y(I)=AUX(N,I)+.2969776*AUX(5,I)+.1587596*Z      HPCG
1975      C      HPCG
1976      Z=X+.4557372*H      HPCG
1977      CALL FCT(Z,Y,DERY)      HPCG
1978      DO 103 I=1,NDIM      HPCG
1979      Z=H*DERY(I)      HPCG
1980      AUX(7,I)=Z      HPCG
1981      103 Y(I)=AUX(N,I)+.2181004*AUX(5,I)-3.050965*AUX(6,I)+3.832865*Z      HPCG
1982      C      HPCG
1983      Z=X+H      HPCG
1984      CALL FCT(Z,Y,DERY)      HPCG
1985      DO 104 I=1,NDIM      HPCG
1986      104 Y(I)=AUX(N,I)+.1747603*AUX(5,I)-.5514807*AUX(6,I)      HPCG
1987      +.1.205536*AUX(7,I)+.1711848*H*DERY(I)      HPCG
1988      GOTO(9,13,15,21),ISW      HPCG
1989      C      HPCG
1990      C      POSSIBLE BREAK-POINT FOR LINKAGE      HPCG
1991      C      HPCG
1992      C      STARTING VALUES ARE COMPUTED.      HPCG
1993      C      NOW START HANNINGS MODIFIED PREDICTOR-CORRECTOR METHOD.      HPCG
1994      200 ISTEP=3      HPCG
1995      201 IF(N-8)204,202,204      HPCG
1996      C      HPCG
1997      C      N=8 CAUSES THE ROWS OF AUX TO CHANGE THEIR STORAGE LOCATIONS      HPCG
1998      202 DO 203 N=2,7      HPCG
1999      DO 203 I=1,NDIM      HPCG
2000      AUX(N-1,I)=AUX(N,I)      HPCG
2001      203 AUX(N+6,I)=AUX(N+7,I)      HPCG
2002      N=7      HPCG
2003      C      HPCG
2004      C      N LESS THAN 8 CAUSES N+1 TO GET N      HPCG
2005      204 N=N+1      HPCG
2006      C      HPCG
2007      C      COMPUTATION OF NEXT VECTOR Y      HPCG
2008      DO 205 I=1,NDIM      HPCG
2009      AUX(N-1,I)=Y(I)      HPCG
2010      205 AUX(N+6,I)=DERY(I)      HPCG
2011      X=X+H      HPCG
2012      206 ISTEP=ISTEP+1      HPCG
2013      DO 207 I=1,NDIM      HPCG
2014      DELTA=AUX(N-4,I)+1.333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+      HPCG
2015      +AUX(N+4,I)+AUX(N+4,I))      HPCG

```

PROGRAM UDKHDEH

```

2016      Y(I)=DELT-.9256198*AUX(16,I)      HPCG
2017      207 AUX(16,I)=DELT      HPCG
2018      C      PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR      HPCG
2019      C      IS GENERATED IN Y. DELT MEANS AN AUXILIARY STORAGE.      HPCG
2020      C      HPCG
2021      CALL FCT(X,Y,DERY)      HPCG
2022      C      DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY      HPCG
2023      C      HPCG
2024      DO 208 I=1,NDIM      HPCG
2025      DELT=.125*(9.*AUX(N-1,I)-AUX(N-3,I)+3.*H*(DERY(I)+AUX(N+6,I)+      HPCG
2026      *AUX(N+6,I)-AUX(N+5,I)))      HPCG
2027      AUX(16,I)=AUX(16,I)-DELT      HPCG
2028      208 Y(I)=DELT+.07438017*AUX(16,I)      HPCG
2029      C      HPCG
2030      C      TEST WHETHER H MUST BE HALVED OR DOUBLED      HPCG
2031      DELT=0.      HPCG
2032      DO 209 I=1,NDIM      HPCG
2033      209 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))      HPCG
2034      IF(DELT-PRMT(4))210,222,222      HPCG
2035      C      HPCG
2036      C      H MUST NOT BE HALVED. THAT MEANS Y(I) ARE GOOD.      HPCG
2037      210 CALL FCT(X,Y,DERY)      HPCG
2038      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)      HPCG
2039      IF(PRMT(5))212,211,212      HPCG
2040      211 IF(IHLF-1)213,212,212      HPCG
2041      212 RETURN      HPCG
2042      213 IF(H*(X-PRMT(2)))214,212,212      HPCG
2043      214 IF(ABS(X-PRMT(2))-.1*ABS(H))212,215,215      HPCG
2044      215 IF(DELT-.02*PRMT(4))216,216,201      HPCG
2045      C      HPCG
2046      C      HPCG
2047      C      H COULD BE DOUBLED IF ALL NECESSARY PRECEDING VALUES ARE      HPCG
2048      C      AVAILABLE      HPCG
2049      216 IF(IHLF)201,201,217      HPCG
2050      217 IF(N-7)201,218,218      HPCG
2051      218 IF(ISTEP-4)201,219,219      HPCG
2052      219 IMOD=ISTEP/2      HPCG
2053      IF(ISTEP-IMOD-IMOD)201,220,201      HPCG
2054      220 H=H+H      HPCG
2055      IHLF=IHLF-1      HPCG
2056      ISTEP=0      HPCG
2057      DO 221 I=1,NDIM      HPCG
2058      AUX(N-1,I)=AUX(N-2,I)      HPCG
2059      AUX(N-2,I)=AUX(N-4,I)      HPCG
2060      AUX(N-3,I)=AUX(N-6,I)      HPCG
2061      AUX(N+6,I)=AUX(N+5,I)      HPCG
2062      AUX(N+5,I)=AUX(N+3,I)      HPCG
2063      AUX(N+4,I)=AUX(N+1,I)      HPCG
2064      DELT=AUX(N+6,I)+AUX(N+5,I)      HPCG
2065      DELT=DELT+DELT+DELT      HPCG
2066      221 AUX(16,I)=8.962963*(Y(I)-AUX(N-3,I))-3.361111*H*(DERY(I)+DELT      HPCG
2067      *AUX(N+4,I))      HPCG
2068      GOTO 201      HPCG
2069      C      HPCG
2070      C      HPCG
2071      C      H MUST BE HALVED      HPCG
2072      222 IHLF=IHLF+1      HPCG
2073      IF(IHLF-10)223,223,210      HPCG
2074      223 H=.5*H      HPCG
2075      ISTEP=0      HPCG
2076      DO 224 I=1,NDIM      HPCG
2077      Y(I)=.00390625*(80.*AUX(N-1,I)+135.*AUX(N-2,I)+40.*AUX(N-3,I)+      HPCG
2078      *AUX(N-4,I))-.1171875*(AUX(N+6,I)-6.*AUX(N+5,I)-AUX(N+4,I))+H      HPCG
2079      AUX(N-4,I)=.0030625*(12.*AUX(N-1,I)+135.*AUX(N-2,I)+      HPCG
2080      *108.*AUX(N-3,I)+AUX(N-4,I))-.0234375*(AUX(N+6,I)+18.*AUX(N+5,I)-

```

PROGRAM UDKHDEM

2081	*9,*AUX(N+4,I)*H	HPCG
2082	AUX(N+3,I)=AUX(N+2,I)	HPCG
2083	224 AUX(N+4,I)=AUX(N+5,I)	HPCG
2084	X=X-H	HPCG
2085	DELT=X-(H+H)	HPCG
2086	CALL FCT(DELT,Y,DERY)	HPCG
2087	DO 225 I=1,NDIM	HPCG
2088	AUX(N+2,I)=Y(I)	HPCG
2089	AUX(N+5,I)=DERY(I)	HPCG
2090	225 Y(I)=AUX(N+4,I)	HPCG
2091	DELT=DELT-(H+H)	HPCG
2092	CALL FCT(DELT,Y,DERY)	HPCG
2093	DO 226 I=1,NDIM	HPCG
2094	DELT=AUX(N+5,I)+AUX(N+4,I)	HPCG
2095	DELT=DELT+DELT+DELT	HPCG
2096	AUX(16,I)=8.962963*(AUX(N+1,I)-Y(I))-3.361111*H*(AUX(N+6,I)+DELT	HPCG
2097	+DERY(I))	HPCG
2098	226 AUX(N+3,I)=DERY(I)	HPCG
2099	GOTO 206	
2100	END	

SECTION 5
UMERGE Listing

PROGRAM UMERGE

```

1      PROGRAM UMERGE
2      C
3      C PROGRAM MERGE--8 JAN 81--BY WALTER E FRICK--MERGING IN CROSS FLOW
4      C
5      C MERGE MODIFIED TO USE UNIVERSAL DATA FILE WITH OPTIONS
6      C
7      C UP-TO-DATE AS OF AUGUST 1985 (NO CHANGES SINCE 6-18-85).
8      C JUNE 18, 1985 REVISED INTERACTIVE DATA INPUT SO USER MAY CORRECT
9      C ERRORS BEFORE RERUNNING THE PROGRAM.
10     C MARCH 27, 1985 CHANGE ANGLE OF DISCHARGE LIMITS FROM 0 TO 90 DEG
11     C TO -5 TO 90 DEG AND DZ CAN BE NEGATIVE IF Z .GE. PDEP
12     C MARCH 19, 1985 RELOCATED THE INTERACTIVE RECYCLE STATEMENT
13     C (3 WRITE) TO 3 CONTINUE, THE FIRST STATEMENT UNDER INTERACTIVE
14     C CONTROL. ADDED STATEMENTS IN SUBROUTINE LIMITS TO DETECT
15     C ERRONEOUS RESPONSES TO PROGRAM PROMPTS I.E. .NE. TO YES OR NO.
16     C
17     C COMMON/OVAR/DPR(30),SAR(30),TAR(30),UAR(30)
18     C COMMON/AVAR/DP(30),SA(30),TA(30),UA(30),DENP(30),DENPP(30)
19     C COMMON/VAR/INTER,IDFP,ICUTOP,IMI,IMO,IAF,IITF,IIFF,INAAF,INABF,
20     C *INACF,QT,NP,PDIA,VANG,PDEP,UW,IR,HANG,SPC,A,ITER,IFRQ,NAA,NAB,NAC,
21     C *NPTS,T,S,DC,FR
22     C COMMON/COUNT/NWO,NW
23     C COMMON/INOUT/IN,IO,IT
24     C COMMON/TVAR/TITLE,TITLI,IFILE
25     C DIMENSION AA(11,90),AB(11,90),AC(11,90),IJ(90),DA(30)
26     C REAL*8 TITLE(10),TITLI(10)
27     C BYTE IFILE(20),OFILE(20)
28     C DATA IFILE(20)/O/,OFILE(20)/O/
29     C
30     C:::::INITIALIZE CONSTANTS AND VARIABLES.
31     C
32     C IN THIS PROGRAM (PDP SYSTEM - FORTRAN IV-PLUS),
33     C LOGICAL UNIT IN IS THE INPUT FILE.
34     C " " IO IS THE OUTPUT FILE.
35     C " " IT IS THE USER TERMINAL.
36     C
37     C IN=1
38     C IO=2
39     C IT=5
40     C
41     C NWO=0
42     C NW=0
43     C G=9.807
44     C ZERO=0.
45     C ONE=1.
46     C TWO=2.
47     C PI=3.1416
48     C BINTO=0.0069555
49     C SMALL=1.0E-6
50     C JCASE=0
51     C TORAD=PI/180.
52     C
53     C OPEN FILES
54     C
55     C WRITE(IT,554)
56     C 554 FORMAT(/1X,' *****PROGRAM UMERGE, AUGUST 1985*****')
57     C WRITE(IT,555)
58     C 555 FORMAT(/1X,' ENTER UDF NAME: ',S)
59     C READ(IT,556)(IFILE(I),I=1,19)
60     C 556 FORMAT(19A1)
61     C WRITE(IT,557)
62     C 557 FORMAT(1X,' ENTER OUTPUT FILE NAME: ',S)
63     C READ(IT,556)(OFILE(I),I=1,19)
64     C OPEN(UNIT=IN,NAME=IFILE,TYPE='OLD')
65     C OPEN(UNIT=IO,NAME=OFILE,TYPE='NEW')

```

PROGRAM UMERGE

```

66 C
67 85 NWO=NWO+NW
68 NCW=0
69 LIMIT=0
70 KPAGE=0
71 JCASE=JCASE+1
72 CALL RMERGE(1,LIMIT)
73 IF(A .LE. SMALL)A=0.1
74 CALL USORT(DPR,SAR,TAR,UAR,NPTS)
75 CALL LIMITS(1,PDEP,VANG,T,S,NPTS,INTER,TITLE,TITLI,
76 *IEXIT,LIMIT)
77 IF(IEXIT .EQ. 0)GO TO 38
78 CALL RMERGE(2,LIMIT)
79 GO TO 85
80 38 IFLAG=1
81 IF(TAR(NPTS) .EQ. 0.)IFLAG=0
82 JFLAG=1
83 IF(T .EQ. 0.)JFLAG=0
84 DO 1000 I=1,NPTS
85 IF(IFLAG .EQ. 0)GO TO 1
86 DENPP(I)=SIGMAT(SAR(I),TAR(I))
87 GO TO 1000
88 1 DENPP(I)=(SAR(I)-1.)*1000.
89 1000 CONTINUE
90 C
91 C IF ONLY ONE DISCHARGE PORT, PORT SPACING IS SET
92 C EQUAL TO 1000. TO PREVENT MERGING PLUME CALCULATIONS.
93 IF(NP .EQ. 1)SPC=1000.
94 C
95 IF(INTER .EQ. 0)GO TO 20
96 C
97 C INTERACTIVE CONTROL
98 C
99 3 CONTINUE
100 IF(NCW .EQ. 0)TQQ=T
101 IF(NCW .EQ. 0)SQ=S
102 IF(NCW .EQ. 0)GO TO 11
103 WRITE(IT,10)(TITLE(IK),IK=1,10)
104 10 FORMAT(/1X,' DO YOU WISH TO CHANGE ANY INPUT VARIABLES AND RERUN?'
105 */1X,' CASE I.D. ',10A8/1X,' YES TO RERUN OR NO FOR NEXT CASE: ',S)
106 READ(IT,9)ANS
107 9 FORMAT(A1)
108 IF(ANS .EQ. 'Y')GO TO 11
109 IF(ANS .EQ. 'N')GO TO 85
110 WRITE(IT,12)
111 12 FORMAT(/1X,' ***** YOU MADE A MISTAKE, TRY AGAIN *****')
112 GO TO 3
113 11 CONTINUE
114 IF(NCW .EQ. 0)WRITE(IT,7)(TITLE(IK),IK=1,10)
115 7 FORMAT(/1X,' CASE IDENTIFICATION:/'
116 *2X,10A8)
117 WRITE(IT,13)
118 13 FORMAT(1X,' WHAT IS THE TITLE OF THIS INTERACTIVE RUN? '/'
119 *2X,S)
120 READ(IT,6)(TITLI(IK),IK=1,10)
121 6 FORMAT(10A8)
122 IF(NCW .EQ. 0)GO TO 39
123 14 CONTINUE
124 WRITE(IT,110)QT,NP,PDIA,SPC,VANG,PDEP
125 110 FORMAT(/1X,' YOUR PRESENT INPUT VARIABLES ARE:/'
126 *1X,' 1. EFFLUENT FLOW (CMS) QT ='F10.4/
127 *1X,' 2. NUMBER OF PORTS NP ='I5/
128 *1X,' 3. PORT DIAMETER (M) PDIA ='F10.4/
129 *1X,' 4. PORT SPACING (M) SPC ='F9.3/
130 *1X,' 5. VERTICAL ANGLE W/HORZ (DEG) VANG ='F9.3/

```

PROGRAM UMERGE

```

131      *1X,' 6. PORT DEPTH (M)                PDEP =',F8.2)
132      IF(TQQ .EQ. 0.)WRITE(IT,109)SQ
133      109 FORMAT(
134      *1X,' 7. EFFLUENT DENSITY (G/CM3)        S =',F11.5)
135      IF(TQQ .NE. 0.)WRITE(IT,112)SQ,TQQ
136      112 FORMAT(
137      *1X,' 7. EFFLUENT SAL (O/OO)            S =',F8.2/
138      *1X,' 8. EFFLUENT TEMP (DEG C)          T =',F8.2)
139      WRITE(IT,113)
140      113 FORMAT(/1X,' ENTER THE NUMBER OF THE VARIABLE
141      * YOU WISH TO CHANGE. ',S)
142      READ(IT,114)NUMBER
143      114 FORMAT(I1)
144      IF(TQQ .EQ. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 7)GO TO 15
145      IF(TQQ .NE. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 8)GO TO 15
146      WRITE(IT,12)
147      GO TO 14
148      15 CONTINUE
149      WRITE(IT,16)
150      16 FORMAT(1X,' WHAT IS THE VALUE OF THIS VARIABLE? ',S)
151      READ(IT,18)VALUE
152      18 FORMAT(F10.0)
153      IF(NUMBER .EQ. 1)QT=VALUE
154      IF(NUMBER .EQ. 2)NP=VALUE
155      IF(NUMBER .EQ. 3)PDIA=VALUE
156      IF(NUMBER .EQ. 4)SPC=VALUE
157      IF(NP .EQ. 1.)SPC=1000.
158      IF(NUMBER .EQ. 5)VANG=VALUE
159      IF(NUMBER .EQ. 6)PDEP=VALUE
160      IF(NUMBER .EQ. 7)SQ=VALUE
161      IF(NUMBER .EQ. 8)TQQ=VALUE
162      111 CONTINUE
163      WRITE(IT,17)
164      17 FORMAT(1X,' DO YOU WISH TO CHANGE ANOTHER VARIABLE?
165      * YES OR NO: ',S)
166      READ(IT,9)ANS
167      IF(ANS .EQ. 'N')GO TO 8
168      IF(ANS .EQ. 'Y')GO TO 14
169      WRITE(IT,12)
170      GO TO 111
171      8 CONTINUE
172      T=TQQ
173      S=SQ
174      CALL LIMITS(2,PDEP,VANG,T,S,NPTS,INTER,TITLE,TITLI,IEXIT,LIMIT)
175      IF(IEXIT .EQ. 0)GO TO 39
176      CALL RMERGE(2,LIMIT)
177      GO TO 85
178      39 NCW=1
179      20 CONTINUE
180      C
181      C      SETUP, AND PRINT INPUT DATA
182      C
183      IF(T .EQ. 0.)DC=(S-1.)*1000.
184      IF(T .NE. 0.)DC=SIGNAT(S,T)
185      B=PDIA*0.5
186      VEL=(QT/NP)/(PI*B*B)
187      U=VEL*COS(VANG+TORAD)
188      V=VEL*SIN(VANG+TORAD)
189      IK=0
190      DO 1100 I=1,NPTS
191      DP(I)=DPR(I)
192      SA(I)=SAR(I)
193      TA(I)=TAR(I)
194      DA(I)=DENPP(I)
195      UA(I)=UAR(I)

```

PROGRAM UMERGE

```

196      DENP(I)=DENPP(I)
197      IF(IK .GT. 0)GO TO 1100
198      IF(DPR(I) .GE. PDEP)IK=I
199  1100 CONTINUE
200      IL=IK
201      IU=IK-1
202      PRODIS=(PDEP-DP(IU))/(DP(IL)-DP(IU))
203      UAA=UA(IU)+PRODIS*(UA(IL)-UA(IU))
204      IF(IFLAG .EQ. 1)GO TO 25
205      DENAS=DA(IU)+PRODIS*(DA(IL)-DA(IU))
206      GO TO 26
207  25 TAA=TA(IU)+PRODIS*(TA(IL)-TA(IU))
208      SAA=SA(IU)+PRODIS*(SA(IL)-SA(IU))
209      DENAS=SIGMAT(SAA,TAA)
210  26 CONTINUE
211      IF(IMI .EQ. 1)GO TO 31
212      GO TO 33
213 C
214  122 CONTINUE
215      IF(KPAGE .EQ. 1 .AND. IMI .EQ. 2)GO TO 29
216      CALL RMERGE(2,LIMIT)
217      WRITE(IO,120)
218  120 FORMAT(1H1,' UMERGE VERSION 1.0 AUGUST 1985.'/2X,4(18(' '),2X))
219      WRITE(IO,125)(IFILE(I),I=1,19)
220  125 FORMAT(1X,' UNIVERSAL DATA FILE: ',19A1)
221      WRITE(IO,121)(TITLE(I),I=1,10)
222  121 FORMAT(2X,'CASE I.D. ',10A8)
223      IF(INTER .EQ. 1)WRITE(IO,202)(TITLI(I),I=1,10)
224      CALL NMERGE(1,IFLAG)
225      KPAGE=INTER
226      GO TO 30
227  29 CONTINUE
228      CALL NMERGE(3,IFLAG)
229  30 IF(IMO .EQ. 2)GO TO 2002
230      WRITE(IO,132)
231  132 FORMAT(/1X,' FIRST LINE OF OUTPUT ARE INITIAL CONDITIONS')
232      WRITE(IO,130)
233  130 FORMAT(/4X,'X',8X,'Z',6X,'PLUME',4X,'DILU-',3X,'DENDIFF',3X,
234      *'HORIZ',4X,'VERT',5X,'TOTAL',4X,'AMBIENT'/18X,'DIAMETER TION'
235      *',15X,'VEL',6X,'VEL',6X,'VEL',5X,'CURRENT'/3X,'(M)',6X,'(M)',6X,
236      *'(M)',12X,'(SIGMAT)',3X,'(M/S)',4X,'(M/S)',4X,'(M/S)',5X,'(M/S)'/)
237      WRITE(IO,400)X,Z,PDIA,DILV,DRHO,U,V,VEL,UW
238      GO TO 2002
239 C
240  31 CONTINUE
241      CALL RMERGE(2,LIMIT)
242      WRITE(IO,131)
243  131 FORMAT(1H1,' UMERGE VERSION 1.0 AUGUST 1985.')
244      WRITE(IO,200)JCASE
245      WRITE(IO,125)(IFILE(I),I=1,19)
246      WRITE(IO,201)(TITLE(I),I=1,10)
247      IF(INTER .EQ. 1)WRITE(IO,202)(TITLI(I),I=1,10)
248      WRITE(IO,203)NPTS,A,ITER,IFRQ,NAA,NAB,NAC
249      IF(IFLAG .NE. 0)GO TO 34
250      WRITE(IO,209)
251  209 FORMAT(/1X,' AMBIENT STRATIFICATION'///' DEPTH SIGMAT
252      * U'/' (M) (M/S)')
253      WRITE(IO,36)(DPR(I),DENPP(I),UAR(I),I=1,NPTS)
254  36 FORMAT(1X,F7.2,F9.2,F9.3)
255      GO TO 33
256  34 WRITE(IO,210)
257  210 FORMAT(/1X,' AMBIENT STRATIFICATION'///' DEPTH SALIN
258      * TEMP SIGMAT U'/' (M) (PPT) (C)
259      * (M/S)')
260      WRITE(IO,37)(DPR(I),SAR(I),TAR(I),DENPP(I),UA(I),I=1,NPTS)

```

PROGRAM UMERGE

```

261      37 FORMAT(1X,F7.2,3F9.2,F9.3)
262      C
263      33 CONTINUE
264      H=B/10.
265      DT=0.01
266      DILV=1.
267      A2=ONE
268      A3=ONE
269      A4=ONE
270      TESMER=ONE
271      II=0
272      PHI=ZERO
273      TAREA=ZERO
274      X=ZERO
275      DB=ZERO
276      DELS=ZERO
277      TIME=ZERO
278      SPCO2=SPC/TWO
279      UO=U
280      VO=V
281      UW=UAA
282      Z=PDEP
283      IF(JFLAG.EQ.1.AND.IFLAG.EQ.1)DELT=T-TAA
284      V1=SQRT(V*V+U*U)
285      VEL=V1
286      DD=VEL*DT
287      DENSFC=1000.+DA(1)
288      DEN=1000.+DC
289      DENA=1000.+DENAS
290      DRHO=DENA-DEN
291      DENNOT=DEN
292      PH=PI*B*B*H/DEN
293      PMO=PH
294      Q=PH/DEN*VEL/H
295      SP=99999.
296      FR=99999.
297      AK=99999.
298      IF(ABS(UW).GT.SMALL)AK=VEL/UW
299      IF(DRHO.GT.SMALL)FR=VEL/SQRT(ABS((DRHO)/DEN)*TWO*B*G)
300      IF(ABS(DENA-DENSFC).GT.SMALL)SP=DEN*PDEP/TWO/B/(DENA
301      *-DENSFC)
302      IF(IMI.NE.1)GO TO 122
303      WRITE(IO,206)AK,FR,Q,SP,PDEP
304      WRITE(IO,205)VEL,UW,B,MP,VANG
305      IF(MP.EQ.1)WRITE(IO,211)
306      IF(MP.GT.1)WRITE(IO,212)SPC
307      WRITE(IO,207)
308      J=0
309      WRITE(IO,208)J,X,Z,PDIA,DILV,U,V,VEL,DRHO,TIME,UW
310      270 FORMAT(1X,' *****NOMINAL TRAPPING LEVEL REACHED')
311      271 FORMAT(1X,' *****MERGING BEGINS')
312      200 FORMAT(2X,123('-')/2X,123('-')/1X,' CASE NUMBER',I5,' WITHIN
313      * THE UDF')
314      201 FORMAT(2X,'CASE I.D.',10A8)
315      202 FORMAT(2X,'RUN TITLE:',10A8)
316      203 FORMAT(/1X,' NPTS=',I3,' A=',F6.3,' ITER=',I5,' IFRQ=',I4/1X,
317      * ' NAA=',I2,' NAB=',I2,' NAC=',I2)
318      206 FORMAT(/
319      *1X,' EFFLUX TO CURRENT RATIO(K) . . .',F10.1/
320      *1X,' DENSIMETRIC FROUDE NO. . . . .',F10.1/
321      *1X,' VOLUME FLUX(M**3/S) . . . . .',F12.3/
322      *1X,' DEPTH AVE STRATIFICATION PARM. .',F10.1/
323      *1X,' DEPTH(M) . . . . .',F11.2)
324      205 FORMAT(
325      *1X,' DISCHARGE VELOCITY(M/S) . . . .',F12.3/

```

PROGRAM UMERGE

```

326      *1X,' CURRENT SPEED(M/S) . . . . .',F12.3/
327      *1X,' PORT RADIUS(M) . . . . .',F13.4/
328      *1X,' NUMBER OF PORTS. . . . .',I8/
329      *1X,' VERTICAL DISCHARGE ANGLE . . . . .',F10.1)
330      211 FORMAT(1X,' PORT SPACING(M). . . . .',1000.0(DEFAULT)')
331      212 FORMAT(1X,' PORT SPACING(M). . . . .',F11.2)
332      207 FORMAT(1X,62(' -')//26X,'MODEL OUTPUT AFTER -J- ITERATIONS
333      *(MKS UNITS) J=0 ARE INITIAL CONDITIONS'//
334      *' J HOR COR(X) DEPTH(Z) DIAMETER VOL DIL HOR VEL(U)
335      * VER VEL(V) TOTAL VEL DEN DIFF TIME CURRENT')
336      208 FORMAT(1X,I4,10F12.3)
337 C
338 C MAIN COMPUTATIONAL SCHEME
339 C
340      2002 IPN=0
341      ITL=0
342      JPN=0
343      JTL=0
344      TT=0
345      SS=0
346      DO 2000 J=1,ITER
347          DO 2001 I=1,MPTS
348              IF(DP(I) .GE. 2)GO TO 40
349      2001 CONTINUE
350      40 LL=I
351          LU=I-1
352          TRATIM=ONE
353              IF(V .GT. ZERO)TRATIM=(DP(LL)-DP(LU))/V
354          TITER=TRATIM/DT
355          BINT=BINTO
356              IF(TITER .LT. 25.)BINT=BINTO+TRATIM/25./DT
357              IF(BINT .LT. BINTO/5.)BINT=BINTO/5.
358          PRODIS=(Z-DP(LU))/(DP(LL)-DP(LU))
359          UW=UA(LU)+PRODIS*(UA(LL)-UA(LU))
360              IF(IFLAG .EQ. 0)GO TO 21
361          TT=TA(LU)+PRODIS*(TA(LL)-TA(LU))
362          SS=SA(LU)+PRODIS*(SA(LL)-SA(LU))
363          RR=SIGMAT(SS,TT)
364          DENA=1000.+RR
365          GO TO 22
366      21 RR=DA(LU)+PRODIS*(DA(LL)-DA(LU))
367          DENA=1000.+RR
368      22 STH=V/VEL
369          CTH=U/VEL
370              IF(B .LE. SPC02)GO TO 41
371          PHI=ATAN(SQRT(B*B-SPC02*SPC02)/SPC02)
372          A2=ONE-TWO*PHI/PI
373          A3=SPC02/B
374          A4=ONE-TWO*PHI/PI+SIN(TWO*PHI)/PI
375      41 CONTINUE
376          CYL=TWO*STH*A3
377          RGRO=PI/DD*CTH*DB*A2
378          CURV=PI/DD*B/TWO*(CTH-UO/V1)*A4
379          EINS=CYL+RGRO+CURV
380          PAREA=EINS*DD*B
381          TAREA=PAREA+TAREA
382          EINS=DENA*JW*DT*B*H*EINS
383              IF(EINS .LT. ZERO)EINS=ZERO
384          UMPER=UW*STH
385          VEN=ABS(VEL-UW*CTH)*A
386          ANGL=ZERO
387              IF(UMPER .GT. VEN)ANGL=ATAN(SQRT(UMPER/VEN-ONE))
388          VEAVE=VEN-ANGL/PI*VEN-UMPER/PI*(ONE-SIN(ANGL))
389          ZWEI=TWO*PI*DENA*H*B*ABS(VEAVE)*A2*DT
390          DM=EINS+ZWEI

```

PROGRAM UMERGE

```

391 GAMMA=DM/BINT/PH
392 DT=DT/GAMMA
393 IF(DT .GT. TWO)GAMMA=GAMMA*DT/TWO
394 IF(DT .GT. TWO)DT=TWO
395 EINS=EINS/GAMMA
396 ZWEI=ZWEI/GAMMA
397 DM=DM/GAMMA
398 TIME=TIME+DT
399 SUM=PM+DM
400 DTH=(STH+UO-CTH+VO)/V1*H/DD
401 TQ=H-ABS(B+DTH)
402 UO=U
403 VO=V
404 U=(PM+U+DM+UN)/SUM
405 IF(JFLAG .EQ. 0 .OR. IFLAG .EQ. 0)GO TO 23
406 T=(PM+T+DM+TT)/SUM
407 S=(PM+S+DM+SS)/SUM
408 DELT=T-TT
409 DELS=S-SS
410 DC=SIGMAT(S,T)
411 DEN=1000.+DC
412 GO TO 24
413 23 DC=(PM+DC+DM+UN)/SUM
414 DEN=1000.+DC
415 24 DRHO=DRHO
416 DRHO=DEN+DEN
417 DENTST=DRHO*DRHO
418 V=PM+V/SUM+DRHO*G/DEN*DT
419 V1=VEL
420 PM=SUM
421 VEL=SQRT(U*U+V*V)
422 DZ=V*DT
423 TESMO=TESMER
424 TESMER=SPC02-B
425 PROMER=TESMER+TESMO
426 DX=U*DT
427 DB=VEL*DT
428 BSAVE=B
429 B=SQRT(PH/(DEN*PI*H))
430 IF(B .GT. SPC02)B=B*SQRT(PI/(PI-TWO*PHI+TWO*COS(PHI)*SIN(PHI)))
431 DB=B*BSAVE
432 H=VEL/V1*H
433 X=X+DX
434 ZOLD=ZD
435 ZD=Z
436 Z=Z-DZ
437 DILZ=DILVO
438 DILVO=DILV
439 DILV=(PM/PMO)*DENNOT/DEN
440 IF(PROMER .LE. ZERO)GO TO 43
441 42 IF(DENTST .LE. ZERO)GO TO 44
442 IF(PROMER .LE. ZERO)GO TO 45
443 46 IF(J .EQ. 1)GO TO 45
444 IF(Z .LE. ZERO)GO TO 45
445 IF(DZ .LE. ZERO .AND. Z .LT. PDEP)GO TO 45
446 IF(J/IFRQ-(J-1)/IFRQ .NE. 1)GO TO 2000
447 GO TO 45
448 43 IF(IPH .NE. 0)GO TO 42
449 IF(ITL .EQ. 0)IPH=1
450 IF(ITL .EQ. 1)IPH=2
451 JPM=J
452 IF(IMO .NE. 2)WRITE(10,271)
453 GO TO 42
454 44 IF(ITL .EQ. 1)GO TO 46
455 JTL=J

```

PROGRAM UMERGE

```

456       IF(IMO .NE. 2)WRITE(10,270)
457       ITL=1
458       XL=-DRHO/(DRHO-DRHO0)
459       TL=ZD+XL*(Z-ZD)
460       SATL=DILVO+XL*(DILV-DILVO)
461 45 DILM=PM/PMO
462       DIA=TWO*B
463       VDIA=DIA*CTH
464       II=II+1
465       IJ(IJ)=J
466       IF(NAA .NE. 1)GO TO 50
467 C     IF IFLAG=0, TT=0
468       AA(1,II)=TT
469       AA(2,II)=VDIA
470       AA(3,II)=H
471       AA(4,II)=PM
472       AA(5,II)=DM
473 C     DELS AND DELT HAVE NO MEANING UNLESS IFLAG=JFLAG=1
474       AA(6,II)=DELS
475       AA(7,II)=DELT
476       AA(8,II)=TAREA
477       AA(9,II)=BINT
478       AA(10,II)=PHI
479       AA(11,II)=ANGL
480 50 CONTINUE
481       IF(NAB .NE. 1)GO TO 51
482       IF(IFLAG .EQ. 1)AB(1,II)=SS
483       IF(IFLAG .EQ. 0)AB(1,II)=DENA
484       AB(2,II)=CTH
485       AB(3,II)=A2
486       AB(4,II)=A3
487       AB(5,II)=A4
488       AB(6,II)=CYL
489       AB(7,II)=RGRO
490       AB(8,II)=CURV
491       AB(9,II)=EINS
492       AB(10,II)=ZWEI
493       AB(11,II)=UW
494 51 CONTINUE
495       IF(NAC .NE. 1)GO TO 52
496       AC(1,II)=PROMER
497       AC(2,II)=DILM
498       AC(3,II)=VEN
499       AC(4,II)=GAMMA
500       AC(5,II)=DT
501 C     IF JFLAG=0, T=0
502       AC(6,II)=T
503       IF(JFLAG .EQ. 1)AC(7,II)=S
504       IF(JFLAG .EQ. 0)AC(7,II)=DEN
505       AC(8,II)=DENTST
506       AC(9,II)=TQ
507       AC(10,II)=DD
508       AC(11,II)=DB
509 52 CONTINUE
510       IF(Z .LE. ZERO)GO TO 70
511       IF(DZ .LE. ZERO .AND. Z .LT. PDEP)GO TO 70
512       IF(IMO .EQ. 1)WRITE(10,208)J,X,Z,DIA,DILV,U,V,VEL,DRHO,TIME,UW
513       IF(IMO .EQ. 0)WRITE(10,400)X,Z,DIA,DILV,DRHO,U,V,VEL,UW
514 400 FORMAT(1X,F6.2,P9.2,P9.3,5P9.2,F10.3)
515 2000 CONTINUE
516       WRITE(10,500)ITER
517 500 FORMAT(1X,' COMPUTATIONS CEASE:  NUMBER OF STEPS EXCEEDS',IS)
518       IF(ITER .EQ. 0)GO TO 85
519       WRITE(17,500)ITER
520       GO TO 3

```

PROGRAM UMERGE

```

521 70 CONTINUE
522 C
523 C PRINT RESULTS
524 C
525 IF(Z .GT. ZERO)GO TO 506
526 DILSUR=DILVO+ZO*((DILVO-DILZ)/(ZOLD-ZO))
527 WRITE(IO,501)
528 506 IF(DZ .LE. ZERO)WRITE(IO,502)
529 IF(JPM .EQ. JTL .AND. JPM .GT. 0)GO TO 503
530 IF(IPM .EQ. 1 .AND. ITL .EQ. 1)WRITE(IO,511)TL,SATL
531 IF(IPM .EQ. 1 .AND. ITL .EQ. 0)WRITE(IO,512)DILSUR
532 IF(IPM .EQ. 0 .AND. ITL .EQ. 1)WRITE(IO,513)TL,SATL
533 IF(IPM .EQ. 0 .AND. ITL .EQ. 0)WRITE(IO,514)DILSUR
534 IF(IPM .EQ. 2)WRITE(IO,515)TL,SATL
535 GO TO 504
536 503 WRITE(IO,516)TL,SATL
537 504 IF(INTER .EQ. 0)GO TO 80
538 IF(Z .LE. ZERO)WRITE(IT,501)
539 IF(DZ .LE. ZERO)WRITE(IT,502)
540 IF(JPM .EQ. JTL .AND. JPM .GT. 0)GO TO 505
541 IF(IPM .EQ. 1 .AND. ITL .EQ. 1)WRITE(IT,511)TL,SATL
542 IF(IPM .EQ. 1 .AND. ITL .EQ. 0)WRITE(IT,512)DILSUR
543 IF(IPM .EQ. 0 .AND. ITL .EQ. 1)WRITE(IT,513)TL,SATL
544 IF(IPM .EQ. 0 .AND. ITL .EQ. 0)WRITE(IT,514)DILSUR
545 IF(IPM .EQ. 2)WRITE(IT,515)TL,SATL
546 GO TO 80
547 505 WRITE(IT,516)TL,SATL
548 80 CONTINUE
549 IF(NAA .EQ. 1)WRITE(IO,301)
550 IF(NAA .EQ. 1)WRITE(IO,300)((IJ(J),(AA(I,J),I=1,11)),J=1,11)
551 IF(NAB .EQ. 1)WRITE(IO,302)
552 IF(NAB .EQ. 1)WRITE(IO,300)((IJ(J),(AB(I,J),I=1,11)),J=1,11)
553 IF(NAC .EQ. 1)WRITE(IO,303)
554 IF(NAC .EQ. 1)WRITE(IO,300)((IJ(J),(AC(I,J),I=1,11)),J=1,11)
555 IF(INTER .EQ. 1)GO TO 3
556 GO TO 85
557 C
558 501 FORMAT(/1X,' COMPUTATIONS CEASE: PLUMES SURFACE')
559 502 FORMAT(/1X,' COMPUTATIONS CEASE: VERTICAL PLUME VELOCITY
560 *IS LESS THAN 0')
561 511 FORMAT(/1X,' PLUMES MERGED BEFORE TRAPPING LEVEL REACHED'/
562 *1X,' TRAPPING LEVEL =',F7.2,' M BELOW SURFACE; DILUTION =',F7.2)
563 512 FORMAT(/1X,' PLUMES MERGED, DID NOT TRAP; DILUTION =',F7.2)
564 513 FORMAT(/1X,' PLUMES DID NOT MERGE'/
565 *1X,' TRAPPING LEVEL =',F7.2,' M BELOW SURFACE; DILUTION =',F7.2)
566 514 FORMAT(/1X,' PLUMES DID NOT MERGE OR TRAP'/
567 *1X,' DILUTION =',F7.2)
568 515 FORMAT(/1X,' PLUMES MERGED AFTER TRAPPING LEVEL REACHED'/
569 *1X,' TRAPPING LEVEL =',F7.2,' M BELOW SURFACE;
570 * DILUTION =',F7.2)
571 516 FORMAT(/1X,' PLUMES MERGED AND TRAPPED AT THE SAME TIME.'/
572 *1X,' TRAPPING LEVEL =',F7.2,' M BELOW SURFACE; DILUTION =',F7.2)
573 300 FORMAT(1X,I4,11E11.3)
574 301 FORMAT('1 J TT VDIA H PM
575 * DM DELS DELT TAREN BINT PHI
576 * ANGL'/)
577 302 FORMAT('1 J SS CTH A2 A3
578 * A4 CYL RGRO CURV EINS ZWEI
579 * UW'/)
580 303 FORMAT('1 J PRMER MASS DIL VEN GAMMA
581 * DT T S DENTST TQ DD
582 * DB'/)
583 END
584 C
585 SUBROUTINE NMERGE(K,IFLAG)

```

PROGRAM UMERGE

```

586 C
587 C PRINTS INPUT DATA FOR UMERGE IN NEW FORMAT
588 C
589 COMMON/OVAR/DPR(30),SAR(30),TAR(30),UAR(30)
590 COMMON/AVAR/DP(30),SA(30),TA(30),UA(30),DENP(30),DENPP(30)
591 COMMON/VAR/INTER,IDFP,ICUTOP,IMI,IMO,IAF,IITF,IIF,INAAF,INABF,
592 *INACF,QT,NP,PDIA,VANG,PDEP,UW,IR,HANG,SPACE,A,ITER,IFRQ,NAA,NAB,
593 *NAC,NPTS,T,S,DC,FR
594 COMMON/COUNT/NWO,NW
595 COMMON/INOUT/IN,IO,IT
596 COMMON/TVAR/TITLE,TITLI,IFILE
597 REAL*8 TITLE(10),TITLI(10)
598 BYTE IFILE(20),OFILE(20)
599 DATA IFILE(20)/O/,OFILE(20)/O/
600 C
601 IF(K-2)1,2,3
602 1 IF(IAF .EQ. 0)WRITE(IO,111)A
603 IF(IAF .EQ. 1)WRITE(IO,112)A
604 IF(IITF .EQ. 0)WRITE(IO,113)ITER
605 IF(IITF .EQ. 1)WRITE(IO,114)ITER
606 IF(IMO .EQ. 2)GO TO 124
607 IF(IIF .EQ. 0)WRITE(IO,115)IFRQ
608 IF(IIF .EQ. 1)WRITE(IO,116)IFRQ
609 124 IF(INAAF .EQ. 0)WRITE(IO,117)NAA
610 IF(INAAF .EQ. 1)WRITE(IO,118)NAA
611 IF(INABF .EQ. 0)WRITE(IO,119)NAB
612 IF(INABF .EQ. 1)WRITE(IO,120)NAB
613 IF(INACF .EQ. 0)WRITE(IO,121)NAC
614 IF(INACF .EQ. 1)WRITE(IO,122)NAC
615 WRITE(IO,4)
616 IF(T .NE. 0)WRITE(IO,132)T
617 IF(T .NE. 0.)WRITE(IO,133)S
618 WRITE(IO,134)DC
619 WRITE(IO,135)FR
620 IF(FLAG .EQ. 0)GO TO 146
621 WRITE(IO,141)
622 WRITE(IO,142)((DPR(I),SAR(I),TAR(I),DENPP(I),UAR(I)),I=1,NPTS)
623 GO TO 2
624 146 WRITE(IO,144)
625 WRITE(IO,143)((DPR(I),DENPP(I),UAR(I)),I=1,NPTS)
626 2 WRITE(IO,200)
627 WRITE(IO,201)QT
628 WRITE(IO,202)NP
629 WRITE(IO,203)PDIA
630 IF(NP .GT. 1)WRITE(IO,131)SPACE
631 IF(NP .EQ. 1)WRITE(IO,123)
632 WRITE(IO,204)VANG
633 WRITE(IO,205)PDEP
634 RETURN
635 3 CONTINUE
636 WRITE(IO,210)(TITLI(IK),IK=1,10)
637 WRITE(IO,135)FR
638 GO TO 2
639 C
640 4 FORMAT(1X)
641 111 FORMAT(/2X,'ASPIRATION ENTRAINMENT COEFFICIENT '='F9.2,4X,
642 *'(DEFAULT)')
643 112 FORMAT(/2X,'ASPIRATION ENTRAINMENT COEFFICIENT '='F9.2)
644 113 FORMAT(2X,'NUMBER OF STEPS ALLOWED',13X,'='I6,7X,'(DEFAULT)')
645 114 FORMAT(2X,'NUMBER OF STEPS ALLOWED',13X,'='I6)
646 115 FORMAT(2X,'ITERATION PRINTOUT FREQUENCY',8X,'='I6,7X,'(DEFAULT)')
647 116 FORMAT(2X,'ITERATION PRINTOUT FREQUENCY',8X,'='I6)
648 117 FORMAT(2X,'PRINT ARRAY AA (0=NO, 1=YES)',8X,'='I6,7X,'(DEFAULT)')
649 118 FORMAT(2X,'PRINT ARRAY AA (0=NO, 1=YES)',8X,'='I6)
650 119 FORMAT(2X,'PRINT ARRAY AB (0=NO, 1=YES)',8X,'='I6,7X,'(DEFAULT)')

```

PROGRAM UMERGE

```

651 120 FORMAT(2X,'PRINT ARRAY AB (0=NO, 1=YES)',8X,'=',I6)
652 121 FORMAT(2X,'PRINT ARRAY AC (0=NO, 1=YES)',8X,'=',I6,7X,'(DEFAULT)')
653 122 FORMAT(2X,'PRINT ARRAY AC (0=NO, 1=YES)',8X,'=',I6)
654 123 FORMAT(2X,'PORT SPACING',24X,'= 1000.0 M(DEFAULT)')
655 131 FORMAT(2X,'PORT SPACING',24X,'=',F9.2,' M')
656 132 FORMAT(2X,'INITIAL TEMPERATURE OF THE PLUME',4X,'=',F9.2,4X,
657 *'DEGREES CENTIGRADE')
658 133 FORMAT(2X,'INITIAL SALINITY OF THE PLUME',7X,'=',F9.2,' PPT')
659 134 FORMAT(2X,'INITIAL DENSITY OF THE PLUME',8X,'=',F11.4,
660 *' SIGMAT UNITS')
661 135 FORMAT(2X,'FROUDE NUMBER',23X,'=',F8.1)
662 141 FORMAT(/3X,'DEPTH',4X,'SALIN',4X,'TEMP',4X,'SIGMAT',6X,'U'/,
663 *4X,'(M)',5X,'(PPT)',5X,'(C)',14X,'(M/S)',/)
664 142 FORMAT(1X,F7.2,3F9.2,F9.3)
665 144 FORMAT(/3X,'DEPTH',3X,'SIGMAT',6X,'U'/4X,'(M)',14X,'(M/S)'/)
666 143 FORMAT(1X,F7.2,F9.2,F9.3)
667 200 FORMAT(1X)
668 201 FORMAT(2X,'TOTAL EFFLUENT FLOW',17X,'=',F11.4,' CMS')
669 202 FORMAT(2X,'NUMBER OF PORTS',21X,'=',I6)
670 203 FORMAT(2X,'PORT DIAMETER',23X,'=',F11.4,' M')
671 204 FORMAT(2X,'VERTICAL PORT ANGLE FROM HORIZONTAL =',F8.1,5X,
672 *'DEGREES')
673 205 FORMAT(2X,'PORT DEPTH',26X,'=',F9.2,' M')
674 210 FORMAT(/2X,70('X'))/2X,'RUN TITLE: ',10A8)
675 END
676 C
677 SUBROUTINE RMERGE(K,LIMIT)
678 C
679 C READ UNIVERSAL DATA FILE AND CONVERT DATA FOR USE IN PROGRAM UMERGE
680 C
681 COMMON/OVAR/DPR(30),SAR(30),TAR(30),UAR(30)
682 COMMON/AVAR/DP(30),SA(30),TA(30),UA(30),DENP(30),DENPP(30)
683 COMMON/VAR/INTER,IDFP,ICUTOP,IMI,IMO,IAF,IITF,IIF,INAAF,INABF,
684 *INACF,QT,NP,PDIA,VANG,PDEP,UW,IR,HANG,SPACE,A,ITER,IFRQ,NAA,NAB,
685 *NAC,NPTS,T,S,DC,FR
686 COMMON/COUNT/NWO,NW
687 COMMON/INOUT/IN,IO,IT
688 COMMON/TVAR/TITLE,TITLI,IFILE
689 REAL*8 TITLE(10),TITLI(10),AA(10)
690 BYTE IFILE(20),OFILE(20)
691 DIMENSION II(5)
692 DATA IFILE(20)/O/,OFILE(20)/O/
693 IF(K.EQ. 2)GO TO 40
694 READ(IN,200,END=99,ERR=999)(TITLE(I),I=1,10)
695 200 FORMAT(10A8)
696 READ(IN,201,ERR=999)INTER,IDFP,ICUTOP,(II(J),J=1,5)
697 201 FORMAT(8I2)
698 READ(IN,202,ERR=999)QT,NP,PDIA,VANG,PDEP
699 202 FORMAT(F10.0,I10,3F10.0)
700 READ(IN,203,ERR=999)UW,HANG,SPACE
701 203 FORMAT(3F10.0)
702 IF(ICUTOP.EQ. 1)READ(IN,213,ERR=999)A,ITER,IFRQ,NAA,NAB,NAC
703 213 FORMAT(F5.0,2I5,3I2)
704 READ(IN,204,ERR=999)NPTS,S,T
705 204 FORMAT(I10,2F10.0)
706 READ(IN,205,ERR=999)((DPR(I),SAR(I),TAR(I),UAR(I)),I=1,NPTS)
707 205 FORMAT(4F10.0)
708 IF(ICUTOP.EQ. 0)GO TO 34
709 IAF=1
710 IITF=1
711 IIF=1
712 INAAF=1
713 INABF=1
714 INACF=1
715 IF(A)21,21,22

```

PROGRAM UMERGE

```

716      21 A=0.1
717      IAF=0
718      22 IF(ITER)23,23,24
719      23 ITER=5000
720      IITF=0
721      24 IF(IFRQ)25,25,26
722      25 IFRQ=150
723      IIFF=0
724      26 IF(NAA)27,27,28
725      27 NAA=0
726      INAAF=0
727      28 IF(NAB)30,30,31
728      30 NAB=0
729      INABF=0
730      31 IF(NAC)32,32,35
731      32 NAC=0
732      INACF=0
733      GO TO 35
734      34 IAF=0
735      IITF=0
736      IIFF=0
737      INAAF=0
738      INABF=0
739      INACF=0
740      A=0.1
741      ITER=5000
742      IFRQ=150
743      NAA=0
744      NAB=0
745      NAC=0
746      35 IMI=II(4)
747      IMO=IMI
748      50 CONTINUE
749      RETURN
750      40 NW=NPTS+5+ICUTOP
751      IF(LIMIT.EQ.0.AND.IDFP.EQ.1)WRITE(IO,403)
752      403 FORMAT(1H1)
753      IF(IDFP.EQ.0)GO TO 50
754      WRITE(IO,400)(IFILE(IK),IK=1,19)
755      400 FORMAT(/1X,' UNIVERSAL DATA FILE: ',19A1/)
756      IF(INTER.EQ.1)WRITE(IO,10)
757      10 FORMAT(
758      *1X,' *****'//
759      *1X,' * NOTE, THIS IS THE ORIGINAL FILE. *'//
760      *1X,' * IT DOES NOT REFLECT CHANGES MADE INTERACTIVELY. *'//
761      *1X,' * THOSE CHANGES ARE SHOWN IN THE OUTPUT HEADING. *'//
762      *1X,' *****'//)
763      REWIND IN
764      IF(NWO.EQ.0)GO TO 41
765      DO 1001 I=1,NWO
766      READ(IN,401)(AA(J),J=1,10)
767      1001 CONTINUE
768      41 DO 1000 I=1,NW
769      READ(IN,401)(AA(J),J=1,10)
770      401 FORMAT(10A8)
771      WRITE(IO,402)(AA(J),J=1,10)
772      1000 CONTINUE
773      402 FORMAT(1X,10A8)
774      GO TO 50
775      999 WRITE(IO,403)
776      WRITE(IO,998)(TITLE(IK),IK=1,10)
777      998 FORMAT(1X,' COMPUTATIONS CEASE FOR: '/2X,'CASE I.D. ',10A8/
778      *1X,' INPUT CONVERSION ERROR, CHECK DATA FILE')
779      IF(INTER.EQ.1)WRITE(IT,998)(TITLE(IK),IK=1,10)
780      99 CALL EXIT

```

PROGRAM UMERGE

```

781      END
782 C
783      SUBROUTINE LIMITS(KK,PDEP,VANG,T,S,NPTS,INTER,TITLE,TITLI,
784      *IEXIT,LIMIT)
785 C
786 C      THIS SUBROUTINE CHECKS LIMITS OF PORT DEPTH (GREATER THAN 0. AND
787 C      LESS THAN OR EQUAL TO PROFILE DEPTH), DISCHARGE ANGLE (EQUAL TO
788 C      OR GREATER THAN -5 DEG BUT LESS THAN OR EQUAL TO 90 DEG,
789 C      VERTICAL, AND IF NEGATIVE, THAT THERE IS AT LEAST ONE PROFILE
790 C      DATA POINT AT LEAST ONE METER BELOW THE PORT DEPTH). THAT THE
791 C      EFFLUENT DENSITY IS LESS THAN THE AMBIENT DENSITY AND THAT THERE
792 C      ARE AMBIENT PROFILE VALUES FOR THE SURFACE, IF INTER=1, THE PORT
793 C      DEPTH, ANGLE AND EFFLUENT DENSITY CAN BE CORRECTED INTERACTIVELY
794 C      BUT SURFACE DATA AND PROFILE DEPTH CORRECTIONS MUST BE MADE TO
795 C      THE DATA SET(S) AND REENTERED. IF INTER=0, ALL CORRECTION MUST BE
796 C      MADE TO THE DATA SET(S) AND REENTERED.
797 C
798      COMMON/OVAR/DPR(30),SAR(30),TAR(30),UAR(30)
799      COMMON/INOUT/IN,IO,IT
800      REAL*8 TITLE(10),TITLI(10)
801      LIMIT=0
802      IDEEP=0
803      IVANG=0
804      NEGANG=0
805      IRHO=0
806      INO=0
807      IEXIT=0
808      100 IF(PDEP .EQ. 0. .OR. DPR(NPTS) .LT. PDEP)GO TO 1
809      10 IF(VANG .LT. 0. .OR. VANG .GT. 90.)GO TO 101
810      30 IF(T .EQ. 0.)RHOE=S
811      IF(T .NE. 0.)RHOE=1.+0.001*SIGMAT(S,T)
812      DO 31 J=2,NPTS
813      IF(PDEP .LE. DPR(J))GO TO 32
814      31 CONTINUE
815      GO TO 40
816      32 JK=J
817      PRO=(PDEP-DPR(JK-1))/(DPR(JK)-DPR(JK-1))
818      TT=TAR(JK-1)+PRO*(TAR(JK)-TAR(JK-1))
819      SS=SAR(JK-1)+PRO*(SAR(JK)-SAR(JK-1))
820      IF(TAR(JK) .EQ. 0.)RHOA=SS
821      IF(TAR(JK) .NE. 0.)RHOA=1.+0.001*SIGMAT(SS,TT)
822      IF(RHOA .GT. RHOE)GO TO 40
823      IRHO=1
824      LIMIT=1
825      IF(INTER .EQ. 0)GO TO 40
826      120 CONTINUE
827      WRITE(IT,33)RHOE,RHOA
828      33 FORMAT(/1X,' EFFLUENT DENSITY ('F7.5,' G/CM3) MUST BE'/
829      *1X,' LESS THAN THE AMBIENT DENSITY ('F7.5,' G/CM3)')
830      WRITE(IT,4)
831      READ(IT,5)NANS
832      IF(NANS .EQ. 'N')GO TO 40
833      IF(NANS .NE. 'Y')WRITE(IT,200)
834      IF(NANS .NE. 'Y')GO TO 120
835      IRHO=2
836      IF(T .EQ. 0.)GO TO 36
837      WRITE(IT,34)
838      34 FORMAT(/1X,' NEW EFFLUENT TEMP VALUE? ',S)
839      READ(IT,7)T
840      WRITE(IT,35)
841      35 FORMAT(1X,' NEW EFFLUENT SAL VALUE? ',S)
842      READ(IT,7)S
843      GO TO 30
844      36 WRITE(IT,37)
845      37 FORMAT(/1X,' NEW EFFLUENT DENSITY VALUE? ',S)

```

PROGRAM UMERGE

```

846      READ(IT,7)S
847      GO TO 30
848      101 IF(VANG .LT. -5. .OR. VANG .GT. 90.)GO TO 11
849      C
850      C:::::IF CONTROL IS NOT PASSED TO 11, VANG .GE. -5. BUT .LT. 0.
851      C      BECAUSE IT'S NEGATIVE, PROFILE DEPTH MUST BE AT LEAST ONE
852      C      METER DEEPER THAN THE PORT DEPTH.
853      C
854      IF(DPR(NPTS)-PDEP .GE. 1.0)GO TO 30
855      NEGANG=1
856      C
857      1 IDEEP=1
858      LIMIT=1
859      IF(INTER .EQ. 0)GO TO 10
860      121 CONTINUE
861      IF(PDEP .EQ. 0.)WRITE(IT,2)
862      2 FORMAT(/1X,' ZERO DISCHARGE DEPTH NOT ALLOWED,
863      * CHANGE PORT DEPTH.')
864      IF(PDEP .GT. 0. .AND. DPR(NPTS) .LT. PDEP)
865      *WRITE(IT,3)PDEP,DPR(NPTS)
866      3 FORMAT(/1X,' PORT DEPTH ('F6.2,' M) IS DEEPER THAN'/
867      *1X,' THE PROFILE DEPTH ('F6.2,' M), CHANGE PORT DEPTH.')
868      IF(NEGANG .EQ. 1)WRITE(IT,21)PDEP,DPR(NPTS)
869      21 FORMAT(/1X,' FOR A NEGATIVE DISCHARGE ANGLE, THE PORT DEPTH ('
870      *F8.4,' M) MUST BE AT'/1X,' LEAST ONE METER LESS THAN THE PROFILE
871      * DEPTH ('F8.4,' M), CHANGE PORT DEPTH.')
872      WRITE(IT,4)
873      4 FORMAT(/1X,' WANT TO MAKE CORRECTION? YES/NO ',S)
874      READ(IT,5)NANS
875      5 FORMAT(A1)
876      IF(NANS .EQ. 'N')GO TO 10
877      IF(NANS .NE. 'Y')WRITE(IT,200)
878      IF(NANS .NE. 'Y')GO TO 121
879      IDEEP=2
880      WRITE(IT,6)
881      6 FORMAT(1X,' NEW VALUE? ',S)
882      READ(IT,7)PDEP
883      7 FORMAT(F7.0)
884      GO TO 100
885      11 IVANG=1
886      LIMIT=1
887      IF(INTER .EQ. 0)GO TO 30
888      122 CONTINUE
889      WRITE(IT,12)VANG
890      12 FORMAT(/1X,' DISCHARGE ANGLE ('F6.1,' ) IS NOT IN THE
891      * RANGE OF -5 TO 90 DEG')
892      WRITE(IT,4)
893      READ(IT,5)NANS
894      IF(NANS .EQ. 'N')GO TO 30
895      IF(NANS .NE. 'Y')WRITE(IT,200)
896      IF(NANS .NE. 'Y')GO TO 122
897      IVANG=2
898      WRITE(IT,6)
899      READ(IT,7)VANG
900      GO TO 10
901      40 IF(DPR(1) .EQ. 0.)GO TO 50
902      INO=1
903      LIMIT=1
904      IF(INTER .EQ. 0)GO TO 50
905      WRITE(IT,41)(TITLE(K),K=1,10)
906      41 FORMAT(1H1/1X,' COMPUTATIONS CEASE FOR'/1X,' CASE I.D. ',10A8)
907      WRITE(IT,44)(TITLI(K),K=1,10)
908      44 FORMAT(1X,' RUN TITLE: ',10A8)
909      WRITE(IT,42)
910      42 FORMAT(/1X,' NO SURFACE DATA FOR THE AMBIENT DENSITY/CURRENT

```

PROGRAM UMERGE

```

911      * PROFILE.'/1X,' WITHOUT THESE DATA, COMPUTATIONAL ERRORS
912      * MAY OCCUR.'/1X,' MAKE CORRECTIONS TO DATA SET(S) AND REENTER.')
913      WRITE(IT,43)
914      43 FORMAT(/1X,' GOING TO NEXT DATA SET IF THERE IS ONE.')
915      50 IF(LIMIT .EQ. 0)GO TO 70
916      IF(INTER .EQ. 1)GO TO 69
917      51 WRITE(IO,41)(TITLE(K),K=1,10)
918      IF(INTER .EQ. 0)GO TO 73
919      IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
920      WRITE(IT,41)(TITLE(K),K=1,10)
921      IF(KK .EQ. 2)WRITE(IT,44)(TITLI(K),K=1,10)
922      WRITE(IT,74)
923      74 FORMAT(/1X,' NECESSARY CORRECTIONS NOT MADE!')
924      WRITE(IT,43)
925      73 WRITE(IO,67)
926      67 FORMAT(/1X,' CORRECT THE FOLLOWING AND REENTER DATA.')
927      IEXIT=1
928      68 IF(IDEEP .EQ. 0)GO TO 52
929      IF(IDEEP .EQ. 1)WRITE(IO,56)PDEP
930      IF(IDEEP .EQ. 2)WRITE(IO,57)PDEP
931      52 IF(IVANG .EQ. 0)GO TO 54
932      IF(IVANG .EQ. 1)WRITE(IO,58)VANG
933      IF(IVANG .EQ. 2)WRITE(IO,59)VANG
934      54 IF(IRHO .EQ. 0)GO TO 55
935      IF(IRHO .EQ. 1)WRITE(IO,64)
936      IF(IRHO .EQ. 2)WRITE(IO,65)RHOE
937      55 IF(INO .EQ. 0)GO TO 75
938      WRITE(IO,42)
939      WRITE(IO,43)
940      GO TO 70
941      69 IF(IDEEP .EQ. 1 .OR. IVANG .EQ. 1 .OR. IRHO .EQ. 1
942      * .OR. INO .EQ. 1)GO TO 51
943      WRITE(IO,66)(TITLE(K),K=1,10)
944      IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
945      WRITE(IO,71)
946      GO TO 68
947      56 FORMAT(
948      *1X,' PORT DEPTH WAS ENTERED AS ',F6.2,' METERS:'/'
949      *1X,' IF THE DISCHARGE ANGLE IS POSITIVE, THE PORT DEPTH MUST'/'
950      *1X,' BE GREATER THAN ZERO BUT LESS THAN OR EQUAL TO THE PROFILE
951      * DEPTH.'/'
952      *1X,' IF THE DISCHARGE ANGLE IS NEGATIVE, THE PORT DEPTH MUST'/'
953      *1X,' BE AT LEAST ONE METER LESS THAN THE PROFILE DEPTH.')
954      57 FORMAT(1X,' PORT DEPTH CHANGED TO: ',F6.2,' M')
955      58 FORMAT(1X,' DISCHARGE ANGLE ('',F6.1,'')
956      * MUST BE .GE. -5 BUT .LE. 90 DEG')
957      59 FORMAT(1X,' DISCHARGE ANGLE CHANGED TO: ',F4.1,' DEG')
958      64 FORMAT(1X,' EFFLUENT DENSITY MUST BE .LT. AMBIENT
959      * DENSITY AT THE DISCHARGE DEPTH')
960      65 FORMAT(1X,' EFFLUENT DENSITY CHANGED TO: ',F7.5,' G/CM3')
961      66 FORMAT(1M1/1X,' COMPUTATIONS CONTINUE FOR'/1X,' CASE I.D. ',10A8)
962      71 FORMAT(1X,' CORRECTIONS WERE INTERACTIVELY MADE TO
963      * THE FOLLOWING:')
964      200 FORMAT(/1X,' *****YOU MADE A MISTAKE, TRY AGAIN*****')
965      75 IF(IEEXIT .EQ. 1)WRITE(IO,43)
966      70 RETURN
967      END
968
969      C
970      SUBROUTINE USORT(DP,X,Y,Z,NPTS)
971      DIMENSION DP(30),X(30),Y(30),Z(30)
972      NESTED=NPTS
973      L=NESTED-1
974      DO 1000 M=1,L
975      NESTED=NESTED-1
976      DO 1000 I=1,NESTED

```

PROGRAM UMERGE

```

976      IF(DP(I) .LE. DP(I+1))GO TO 1000
977      DUMMY=DP(I)
978      DP(I)=DP(I+1)
979      DP(I+1)=DUMMY
980      DUMMY=X(I)
981      X(I)=X(I+1)
982      X(I+1)=DUMMY
983      DUMMY=Y(I)
984      Y(I)=Y(I+1)
985      Y(I+1)=DUMMY
986      DUMMY=Z(I)
987      Z(I)=Z(I+1)
988      Z(I+1)=DUMMY
989      1000 CONTINUE
990      RETURN
991      END
992  C
993      FUNCTION SIGMAT(SAL,T)
994      SIGO=((6.8E-6*SAL)-4.82E-4)*SAL+.8149)*SAL-0.093
995      B=1.E-6*T*((.01667*T-.8164)*T+18.03)
996      A=.001*T*((.0010843*T-.09818)*T+4.7867)
997      SUMT=(T-3.98)*(T-3.98)*(T+283.)/(503.57*(T+67.26))
998      SIGMAT=(SIGO+.1324)*(1.-A+B*(SIGO-.1324))-SUMT
999      RETURN
1000     END

```

SECTION 6
ULINE Listing

PROGRAM ULINE

```

1      PROGRAM ULINE
2      C
3      C      UP-TO-DATE AS OF AUGUST 1985 (NO CHANGE SINCE 6-18-85).
4      C      JUNE 18, 1985 REVISED INTERACTIVE DATA INPUT SO USER MAY CORRECT
5      C      ERRORS BEFORE RERUNNING THE PROGRAM.
6      C      MARCH 19, 1985 RELOCATED THE INTERACTIVE RECYCLE STATEMENT
7      C      (3 WRITE) TO 3 CONTINUE, THE FIRST STATEMENT UNDER INTERACTIVE
8      C      CONTROL. ADDED STATEMENTS IN SUBROUTINE LIMITS TO DETECT
9      C      ERRONEOUS RESPONSES TO PROGRAM PROMPTS I.E. .NE. TO YES OR NO.
10     C
11     C      THE DIFFUSER IS APPROXIMATED AS A LINE SOURCE OF BUOYANCY
12     C      FLUX ONLY.
13     C
14     C      SUBROUTINES USORT, LIMITS, SIGMAT, CALC, RLINE, NLINE
15     C
16     C      COMMON/AVAR/DP(30),SA(30),TA(30),UA(30),DENP(30)
17     C      COMMON/VAR/INTER,IDFP,ILI,ILO,IRKF,QT,NP,PDEP,U,SPACE,NPTS,T,S,
18     C      *HANG,RHOE,RK,DH,IDH,FR
19     C      COMMON/COUNT/NWO,NW
20     C      COMMON/INOUT/IN,IO,IT
21     C      DIMENSION D(30),RHO(30),AR(30),BR(30),HH(30),AU(30),BU(30)
22     C      REAL*8 TITLE(10),TITLI(10)
23     C      BYTE IFILE(20),OFILE(20)
24     C      COMMON/TVAR/TITLE,TITLI,IFILE
25     C      DATA IFILE(20)/0/,OFILE(20)/0/
26     C
27     C      IN THIS PROGRAM (PDP SYSTEM - FORTRAN IV-PLUS),
28     C      LOGICAL UNIT IN IS THE INPUT FILE.
29     C      "      " IO IS THE OUTPUT FILE.
30     C      "      " IT IS THE USER TERMINAL.
31     C
32     C      IN=1
33     C      IO=2
34     C      IT=5
35     C      NWO=0
36     C      NW=0
37     C      T13=1./3.
38     C
39     C      OPEN FILES
40     C
41     C      WRITE(IT,554)
42     C      554 FORMAT(/1X,' *****PROGRAM ULINE, AUGUST 1985*****'/
43     C      *1X,' A LINE SOURCE OF BUOYANCY FLUX ONLY')
44     C      WRITE(IT,555)
45     C      555 FORMAT(/1X,' ENTER UDF NAME: ',S)
46     C      READ(IT,556)(IFILE(I),I=1,19)
47     C      556 FORMAT(19A1)
48     C      WRITE(IT,557)
49     C      557 FORMAT(1X,' ENTER OUTPUT FILE NAME: ',S)
50     C      READ(IT,556)(OFILE(I),I=1,19)
51     C      OPEN(UNIT=IN,NAME=IFILE,TYPE='OLD')
52     C      OPEN(UNIT=IO,NAME=OFILE,TYPE='NEW')
53     C
54     C      85 NWO=NWO+NW
55     C      NCW=0
56     C      LIMIT=0
57     C      CALL RLINE(1,LIMIT)
58     C      CALL USORT(DP,SA,TA,UA,NPTS)
59     C      CALL LIMITS(1,NP,PDEP,HANG,T,S,NPTS,
60     C      *INTER,TITLE,TITLI,IEXIT,LIMIT)
61     C      IF(IEXIT.EQ. 0)60 TO 50
62     C      CALL RLINE(2,LIMIT)
63     C      GO TO 85
64     C      50 IF(T)52,51,52
65     C      51 KFLAG=0

```

PROGRAM ULINE

```

66      RHOE=S
67      GO TO 53
68      52 KFLAG=1
69      RHOE=1.+0.001*SIGMAT(S,T)
70      53 IF(TA(NPTS))55,54,55
71      54 IFLAG=0
72      GO TO 1
73      55 IFLAG=1
74      DO 1000 I=1,NPTS
75          DENP(I)=1.+0.001*SIGMAT(SA(I),TA(I))
76      1000 CONTINUE
77      GO TO 2
78      1 DO 1001 I=1,NPTS
79          DENP(I)=SA(I)
80      1001 CONTINUE
81      2 KPAGE=0
82          IF(INTER .EQ. 0)GO TO 20
83      C
84      C INTERACTIVE CONTROL
85      C
86      3 CONTINUE
87          IF(NCW .EQ. 0)GO TO 11
88          WRITE(IT,10)(TITLE(IK),IK=1,10)
89      10 FORMAT(/1X,' DO YOU WISH TO CHANGE ANY INPUT VARIABLES AND RERUN?'
90          */1X,' CASE I.D. ',10A8/1X,' YES TO RERUN OR NO FOR NEXT CASE: ',S)
91          READ(IT,9)ANS
92      9 FORMAT(A1)
93          IF(ANS .EQ. 'Y')GO TO 11
94          IF(ANS .EQ. 'N')GO TO 85
95          WRITE(IT,12)
96      12 FORMAT(/1X,' ***** YOU MADE A MISTAKE, TRY AGAIN *****')
97          GO TO 3
98      11 CONTINUE
99          IF(NCW .EQ. 0)WRITE(IT,7)(TITLE(IK),IK=1,10)
100      7 FORMAT(/1X,' CASE IDENTIFICATION:/'
101          *2X,10A8)
102          WRITE(IT,13)
103      13 FORMAT(1X,' WHAT IS THE TITLE OF THIS INTERACTIVE RUN? '/'
104          *2X,S)
105          READ(IT,6)(TITLI(IK),IK=1,10)
106      6 FORMAT(10A8)
107          IF(NCW .EQ. 0)GO TO 19
108      14 CONTINUE
109          WRITE(IT,100)QT,NP,SPACE,HANG,PDEP
110      100 FORMAT(/1X,' YOUR PRESENT INPUT VARIABLES ARE:'''
111          *1X,' 1. EFFLUENT FLOW (CMS)          QT ='F10.4/
112          *1X,' 2. NUMBER OF PORTS              NP ='F5/
113          *1X,' 3. PORT SPACING (M)            SPACE ='F9.3/
114          *1X,' 4. HORZ ANGLE OF THE CURRENT'/'
115          *1X,' WITH THE DIFFUSER (DEG)        HANG ='F9.2/
116          *1X,' 5. PORT DEPTH (M)              PDEP ='F8.2)
117          IF(T .EQ. 0.)WRITE(IT,99)S
118      99 FORMAT(
119          *1X,' 6. EFFLUENT DENSITY (G/CM3)      S ='F11.5)
120          IF(T .NE. 0.)WRITE(IT,101)S,T
121      101 FORMAT(
122          *1X,' 6. EFFLUENT SAL (O/OO)          S ='F8.2/
123          *1X,' 7. EFFLUENT TEMP (DEG C)        T ='F8.2)
124          WRITE(IT,21)
125      21 FORMAT(/1X,' ENTER THE NUMBER OF THE VARIABLE
126          * YOU WISH TO CHANGE. ',S)
127          READ(IT,22)NUMBER
128      22 FORMAT(I1)
129          IF(T .EQ. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 6)GO TO 15
130          IF(T .NE. 0. .AND. NUMBER .GE. 1 .AND. NUMBER .LE. 7)GO TO 15

```

PROGRAM ULINE

```

131      WRITE(IT,12)
132      GO TO 14
133  15 CONTINUE
134      WRITE(IT,16)
135  16 FORMAT(1X,' WHAT IS THE VALUE OF THIS VARIABLE? ',S)
136      READ(IT,18)VALUE
137  18 FORMAT(F10.0)
138      IF(NUMBER .EQ. 1)QT=VALUE
139      IF(NUMBER .EQ. 2)NP=VALUE
140      IF(NUMBER .EQ. 3)SPACE=VALUE
141      IF(NUMBER .EQ. 4)HANG=VALUE
142      IF(NUMBER .EQ. 5)PDEP=VALUE
143      IF(NUMBER .EQ. 6)S=VALUE
144      IF(NUMBER .EQ. 7)T=VALUE
145  23 CONTINUE
146      WRITE(IT,17)
147  17 FORMAT(1X,' DO YOU WISH TO CHANGE ANOTHER VARIABLE?
148      * YES OR NO: ',S)
149      READ(IT,9)ANS
150      IF(ANS .EQ. 'N')GO TO 8
151      IF(ANS .EQ. 'Y')GO TO 14
152      WRITE(IT,12)
153      GO TO 23
154  8 CONTINUE
155      CALL LIMITS(2,NP,PDEP,HANG,T,S,NPTS,INTER,TITLE,TITLI,IEXIT,LIMIT)
156      IF(IEXIT .EQ. 0)GO TO 19
157      CALL RLINE(2,LIMIT)
158      GO TO 85
159  19 NCV=1
160  20 THETA=HANG
161      IF(HANG .GT. 90.)THETA=180.-HANG
162  C
163  C      MAIN COMPUTATIONAL SCHEME
164  C
165      N=NPTS
166      DO 1010 I=1,NPTS
167          D(I)=DP(I)
168          RHO(I)=DENP(I)
169          IF(PDEP .GT. D(I))GO TO 1010
170          D(I)=PDEP
171          RHO(I)=(DENP(I)-DENP(I-1))/(DP(I)-DP(I-1))*
172          *(PDEP-DP(I-1))+DENP(I-1)
173          N=I
174          GO TO 30
175  1010 CONTINUE
176      30 NM1=N-1
177          QL=QT/(SPACE*(NP-1))
178          DO 1100 I=1,NM1
179              II=N-I
180              AR(II)=RHO(I+1)
181              BR(II)=(RHO(I)-RHO(I+1))/(D(I)-D(I+1))
182              AU(II)=UA(I+1)
183              BU(II)=(UA(I)-UA(I+1))/(D(I)-D(I+1))
184              HH(II)=D(N)-D(I+1)
185  1100 CONTINUE
186          HH(N)=D(N)
187          AR(N)=RHO(1)
188          G=9.81
189          GP=G*(AR(1)-RHOE)/RHOE
190          RB3=(GP*QL)**T13
191          IF(THETA .GT. 45.)GO TO 35
192          F1=(45.-THETA)/45.
193          K1=1
194          GO TO 36
195  35 F1=(90.-THETA)/45.

```

PROGRAM ULINE

```

196      K1=2
197 36 K2=K1+1
198      K=0
199      U=AU(1)
200      F=U*U/(GP*QL)
201      FR=F
202      IF(U .EQ. 0.) ALPH=RK*0.27*RB3/QL
203      IF(U .GT. 0.) ALPH=RK*U*(F1*CALC(F,K1)+(1.-F1)*CALC(F,K2))/QL
204      SUM=0.
205      SUM1=0.
206      HI=0
207      RI=AR(1)
208      DO 2000 I=1,NH1
209          XNH=(NH(I+1)-NH(I))/DH
210          IF(XNH .GT. 500.) XNH=500.
211          IF(XNH .LT. 1.) XNH=1.
212          NH=XNH
213          DHI=(NH(I+1)-NH(I))/NH
214          DH2=DHI/2.
215          DO 2000 J=1,NH
216              K=K+1
217              HI1=HI+DHI
218              RI1=AR(I)+BR(I)*(HI1-NH(I))
219              U=AU(I)+BU(I)*(HI1-NH(I))
220              IF(U) 37,37,38
221 37      ALPH1=RK*0.27*RB3/QL
222              GO TO 39
223 38      F=U*U/(GP*QL)
224              FF=F1*CALC(F,K1)+(1.-F1)*CALC(F,K2)
225              ALPH1=RK*U*FF/QL
226 39      SUM=SUM+(ALPH*RI+ALPH1*RI1)*DH2
227              SUM1=SUM1+(ALPH+ALPH1)*DH2
228              RPI1=(RHOE+(SUM/SUM1)*(SUM1-1))/SUM1
229              IF(RPI1-RI1) 40,45,45
230 40      SUM=SUM1
231              ALPH=ALPH1
232              RI=RI1
233              HI=HI1
234              RPI=RPI1
235 2000 CONTINUE
236      TL=0
237      SAA=SUM1
238      ISUR=1
239      GO TO 46
240 45 A1=(RI1-RI)/DHI
241      B1=RI
242      A2=(RPI1-RPI)/DHI
243      B2=RPI
244      Z=(B2-B1)/(A1-A2)/DHI
245      TL=NH(N)-(HI+Z*DHI)
246      SAA=SUM*(1.-Z)+SUM1*Z
247      ISUR=0
248 46 CONTINUE
249 C
250 C      PRINT FINAL RESULTS
251 C
252      IF(KPAGE .EQ. 1 .AND. ILI .EQ. 2) GO TO 61
253      CALL RLINE(2,LIMIT)
254      WRITE(IO,201)
255 201 FORMAT(1H1,1X,'ULINE VERSION 2.0 AUGUST 1985
256      *   A LINE SOURCE OF BUOYANCY FLUX ONLY'/2X,4(15(' '),2X))
257      WRITE(IO,205)(FILE(IK),IK=1,19)
258 205 FORMAT(1X,' UNIVERSAL DATA FILE: ',19A1)
259      WRITE(IO,202)(TITLE(IK),IK=1,10)
260 202 FORMAT(2X,'CASE I.D. ',10A8)

```

PROGRAM ULINE

```

261      IF(INTER .EQ. 1)WRITE(10,203)(TITLI(IK),IK=1,10)
262 203 FORMAT(2X,'RUN TITLE: ',10A8)
263      CALL NLINE(1,IFLAG)
264      GO TO 62
265      61 CALL NLINE(3,IFLAG)
266      62 IF(ISUR .EQ. 0)WRITE(10,210)TL,SAA
267 210 FORMAT(/1X,' TRAPPING LEVEL =',F7.2,' M BELOW WATER SURFACE, '
268      *,'DILUTION =',F7.2)
269      IF(ISUR .EQ. 1)WRITE(10,211)SAA
270 211 FORMAT(/1X,' PLUME SURFACES, DILUTION =',F7.2)
271      IF(INTER .EQ. 0)GO TO 85
272      IF(ISUR .EQ. 0)WRITE(10,210)TL,SAA
273      IF(ISUR .EQ. 1)WRITE(10,211)SAA
274      KPAGE=1
275      GO TO 3
276      END
277 C
278      SUBROUTINE LIMITS(KK,MP,PDEP,HANG,T,S,NPTS,INTER,TITLE,TITLI,
279      *IEXIT,LIMIT)
280 C
281 C      THIS SUBROUTINE CHECKS LIMITS OF PORT DEPTH (GREATER THAN 0. AND
282 C      LESS THAN OR EQUAL TO PROFILE DEPTH), THAT THERE ARE TWO OR MORE
283 C      PORTS, CURRENT ANGLE RELATIVE TO THE DIFFUSER (EQUAL TO OR GREATER
284 C      THAN 0. BUT LESS THAN OR EQUAL TO 180. DEG., 90 DEG. IS PERPEN-
285 C      DICULAR TO THE DIFFUSER, WITHIN THE PROGRAM THIS ANGLE IS FURTHER
286 C      LIMITED TO 0 - 90 DEG.), THAT THE EFFLUENT DENSITY IS EQUAL TO OR
287 C      LESS THAN THE AMBIENT DENSITY AND THAT THERE ARE AMBIENT PROFILE
288 C      VALUES FOR THE SURFACE, I.E. DP(1)=0. IF INTER=1, THE PORT DEPTH,
289 C      NUMBER OF PORTS, ANGLE AND EFFLUENT DENSITY CAN BE CORRECTED
290 C      INTERACTIVELY BUT SURFACE DATA CORRECTION MUST BE MADE TO THE
291 C      DATA SET(S) AND REENTERED. IF INTER=0, ALL CORRECTION MUST BE MADE
292 C      TO THE DATA SET(S) AND REENTERED.
293 C
294      COMMON/AVAR/DP(30),SA(30),TA(30),UA(30),DENP(30)
295      COMMON/INOUT/IN,IO,IT
296      REAL*8 TITLE(10),TITLI(10)
297      LIMIT=0
298      IDEEP=0
299      IHANG=0
300      IRHO=0
301      INO=0
302      INP=0
303      IEXIT=0
304 100 IF(PDEP .EQ. 0. .OR. DP(NPTS) .LT. PDEP)GO TO 1
305      10 IF(HANG .LT. 0. .OR. HANG .GT. 180.)GO TO 11
306      20 IF(NP .LT. 2)GO TO 21
307      30 IF(T .EQ. 0.)RHOE=S
308      IF(T .NE. 0.)RHOE=1.+0.001*SIGMAT(S,T)
309      DO 31 J=2,NPTS
310      IF(PDEP .LE. DP(J))GO TO 32
311      31 CONTINUE
312      GO TO 40
313      32 JK=J
314      PRO=(PDEP-DP(JK-1))/(DP(JK)-DP(JK-1))
315      TT=TA(JK-1)+PRO*(TA(JK)-TA(JK-1))
316      SS=SA(JK-1)+PRO*(SA(JK)-SA(JK-1))
317      IF(TA(JK) .EQ. 0.)RHOA=SS
318      IF(TA(JK) .NE. 0.)RHOA=1.+0.001*SIGMAT(SS,TT)
319      IF(RHOA .GE. RHOE)GO TO 40
320      IRHO=1
321      LIMIT=1
322      IF(INTER .EQ. 0)GO TO 40
323 120 CONTINUE
324      WRITE(10,33)RHOE,RHOA
325 33 FORMAT(/1X,' EFFLUENT DENSITY (' ,F7.5,' G/CM3) IS GREATER'/

```

PROGRAM ULINE

```

326      *1X,' THAN THE AMBIENT DENSITY ('F7.5,' G/CM3')
327      WRITE(IT,4)
328      READ(IT,5)NANS
329      IF(NANS .EQ. 'N')GO TO 40
330      IF(NANS .NE. 'Y')WRITE(IT,200)
331      IF(NANS .NE. 'Y')GO TO 120
332      IRHO=2
333      IF(T .EQ. 0.)GO TO 36
334      WRITE(IT,34)
335      34 FORMAT(/1X,' NEW TEMP VALUE? ',S)
336      READ(IT,7)T
337      WRITE(IT,35)
338      35 FORMAT(1X,' NEW SAL VALUE? ',S)
339      READ(IT,7)S
340      GO TO 30
341      36 WRITE(IT,37)
342      37 FORMAT(/1X,' NEW DENSITY VALUE? ',S)
343      READ(IT,7)S
344      GO TO 30
345      1 IDEEP=1
346      LIMIT=1
347      IF(INTER .EQ. 0)GO TO 10
348      121 CONTINUE
349      IF(PDEP .EQ. 0.)WRITE(IT,2)
350      2 FORMAT(/1X,' ZERO DISCHARGE DEPTH NOT ALLOWED.')
351      IF(PDEP .GT. 0.)WRITE(IT,3)PDEP,DP(NPTS)
352      3 FORMAT(/1X,' PORT DEPTH ('F6.2,' M) IS DEEPER THAN'/
353      *1X,' THE PROFILE DEPTH ('F6.2,' M).')
354      WRITE(IT,4)
355      4 FORMAT(/1X,' WANT TO MAKE CORRECTION? YES/NO ',S)
356      READ(IT,5)NANS
357      5 FORMAT(A1)
358      IF(NANS .EQ. 'N')GO TO 10
359      IF(NANS .NE. 'Y')WRITE(IT,200)
360      IF(NANS .NE. 'Y')GO TO 121
361      IDEEP=2
362      WRITE(IT,6)
363      6 FORMAT(1X,' NEW VALUE? ',S)
364      READ(IT,7)PDEF
365      7 FORMAT(F7.0)
366      GO TO 100
367      11 IHANG=1
368      LIMIT=1
369      IF(INTER .EQ. 0)GO TO 30
370      122 CONTINUE
371      WRITE(IT,12)
372      12 FORMAT(/1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER
373      * IS NOT IN THE RANGE OF 0 TO 180 DEG')
374      WRITE(IT,4)
375      READ(IT,5)NANS
376      IF(NANS .EQ. 'N')GO TO 30
377      IF(NANS .NE. 'Y')WRITE(IT,200)
378      IF(NANS .NE. 'Y')GO TO 122
379      IHANG=2
380      WRITE(IT,6)
381      READ(IT,7)HANG
382      GO TO 10
383      21 INP=1
384      LIMIT=1
385      IF(INTER .EQ. 0)GO TO 30
386      123 CONTINUE
387      WRITE(IT,22)
388      22 FORMAT(/1X,' MODEL REQUIRES TWO OR MORE PORTS')
389      WRITE(IT,4)
390      READ(IT,5)NANS

```

PROGRAM ULINE

```

391 IF(NANS .EQ. 'N')GO TO 30
392 IF(NANS .NE. 'Y')WRITE(IT,200)
393 IF(NANS .NE. 'Y')GO TO 123
394 INP=2
395 WRITE(IT,6)
396 READ(IT,23)NP
397 23 FORMAT(I4)
398 GO TO 20
399 40 IF(OP(1) .EQ. 0.)GO TO 50
400 INO=1
401 LIMIT=1
402 IF(INTER .EQ. 0)GO TO 50
403 WRITE(IT,41)(TITLE(K),K=1,10)
404 41 FORMAT(1H1/1X,' COMPUTATIONS CEASE FOR'/1X,' CASE I.D. ',10A8)
405 WRITE(IT,44)(TITLI(K),K=1,10)
406 44 FORMAT(1X,' RUN TITLE: ',10A8)
407 WRITE(IT,42)
408 42 FORMAT(/1X,' NO SURFACE DATA FOR THE AMBIENT DENSITY/CURRENT
409 * PROFILE.'/1X,' WITHOUT THESE DATA, COMPUTATIONAL ERRORS
410 * MAY OCCUR.'/1X,' MAKE CORRECTIONS TO DATA SET(S) AND REENTER.')
411 WRITE(IT,43)
412 43 FORMAT(/1X,' GOING TO NEXT DATA SET IF THERE IS ONE.')
413 50 IF(LIMIT .EQ. 0)GO TO 70
414 IF(INTER .EQ. 1)GO TO 69
415 51 WRITE(IO,41)(TITLE(K),K=1,10)
416 IF(INTER .EQ. 0)GO TO 73
417 IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
418 WRITE(IT,41)(TITLE(K),K=1,10)
419 IF(KK .EQ. 2)WRITE(IT,44)(TITLI(K),K=1,10)
420 WRITE(IT,74)
421 74 FORMAT(/1X,' NECESSARY CORRECTIONS NOT MADE:')
422 WRITE(IT,43)
423 73 WRITE(IO,67)
424 67 FORMAT(/1X,' CORRECT THE FOLLOWING AND REENTER DATA.')
425 IEXIT=1
426 68 IF(IDEEP .EQ. 0)GO TO 52
427 IF(IDEEP .EQ. 1)WRITE(IO,56)PDEP
428 IF(IDEEP .EQ. 2)WRITE(IO,57)PDEP
429 52 IF(IHANG .EQ. 0)GO TO 53
430 IF(IHANG .EQ. 1)WRITE(IO,58)
431 IF(IHANG .EQ. 2)WRITE(IO,59)VANG
432 53 IF(INP .EQ. 0)GO TO 54
433 IF(INP .EQ. 1)WRITE(IO,62)
434 IF(INP .EQ. 2)WRITE(IO,63)NP
435 54 IF(IRHO .EQ. 0)GO TO 55
436 IF(IRHO .EQ. 1)WRITE(IO,64)
437 IF(IRHO .EQ. 2)WRITE(IO,65)RHOE
438 55 IF(INO .EQ. 0)GO TO 75
439 WRITE(IO,42)
440 WRITE(IO,43)
441 GO TO 70
442 69 IF(IDEEP .EQ. 1 .OR. IHANG .EQ. 1 .OR. IRHO .EQ. 1
443 *.OR. INO .EQ. 1 .OR. INP .EQ. 1)GO TO 51
444 WRITE(IO,66)(TITLE(K),K=1,10)
445 IF(KK .EQ. 2)WRITE(IO,44)(TITLI(K),K=1,10)
446 WRITE(IO,71)
447 GO TO 68
448 56 FORMAT(
449 *1X,' PORT DEPTH WAS ENTERED AS ',F6.2,' METERS:/'
450 *1X,' THE PORT DEPTH MUST BE GREATER THAN ZERO BUT/'
451 *1X,' LESS THAN OR EQUAL TO THE PROFILE DEPTH.')
452 57 FORMAT(1X,' PORT DEPTH CHANGED TO: ',F6.2,' M')
453 58 FORMAT(1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER
454 * MUST BE .GE. 0 BUT .LE. 180 DEG')
455 59 FORMAT(1X,' CURRENT ANGLE RELATIVE TO THE DIFFUSER

```

PROGRAM ULINE

```

456      * CHANGED TO: ',F6.2,' DEG')
457      62 FORMAT(1X,' MODEL REQUIRES TWO OR MORE PORTS')
458      63 FORMAT(1X,' NUMBER OF PORTS CHANGED TO: ',I4)
459      64 FORMAT(1X,' EFFLUENT DENSITY MUST BE .LE. AMBIENT
460      * DENSITY AT THE DISCHARGE DEPTH')
461      65 FORMAT(1X,' EFFLUENT DENSITY CHANGED TO: ',F7.5,' G/CM3')
462      66 FORMAT(1H1/1X,' COMPUTATIONS CONTINUE FOR'/1X,' CASE I.D. ',10A8)
463      71 FORMAT(1X,' CORRECTIONS WERE INTERACTIVELY MADE TO
464      * THE FOLLOWING: '/')
465      200 FORMAT(/1X,' *****YOU MADE A MISTAKE, TRY AGAIN*****')
466      75 IF(IEXIT .EQ. 1)WRITE(10,43)
467      70 RETURN
468      END
469  C
470      SUBROUTINE RLINE(K,LIMIT)
471  C
472  C      READ UNIVERSAL DATA FILE AND CONVERT DATA FOR USE IN PROGRAM ULINE
473  C
474      COMMON/AVAR/DP(30),SA(30),TA(30),UA(30),DENP(30)
475      COMMON/VAR/INTER,IDFP,ILI,ILO,IRKF,QT,NP,PDEP,U,SPACE,NPTS,T,S,
476      *HANG,RHOE,RK,DH,IDH,FR
477      COMMON/SAV/II(5),ICUTOP
478      COMMON/COUNT/NWO,NW
479      COMMON/INOUT/IN,IO,IT
480      REAL*8 TITLE(10),TITLI(10),AA(10)
481      BYTE IFILE(20),OFILE(20)
482      COMMON/TVAR/TITLE,TITLI,IFILE
483      DATA IFILE(20)/O/,OFILE(20)/O/
484  C
485      IF(K .EQ. 2)GO TO 40
486      DH=0.
487      READ(IN,200,END=99,ERR=999)(TITLE(I),I=1,10)
488      200 FORMAT(10A8)
489      READ(IN,201,ERR=999)INTER,IDFP,ICUTOP,(II(J),J=1,5)
490      201 FORMAT(8I2)
491      READ(IN,202,ERR=999)QT,NP,PDIA,VANG,PDEP
492      202 FORMAT(F10.0,I10,3F10.0)
493      READ(IN,203,ERR=999)UW,HANG,SPACE
494      203 FORMAT(3F10.0)
495  C
496  C      ALL *X* VARIABLES ARE IGNORED
497  C
498      IF(ICUTOP .EQ. 1)READ(IN,213,ERR=999)AX,IX1,IX2,IX3,IX4,IX5,
499      *PX,RK,DH,HX,EX,AA,IX6,IX7
500  C
501  C
502      213 FORMAT(F5.0,2I5,3I2,6F5.0,2I5)
503      READ(IN,204,ERR=999)NPTS,S,T
504      204 FORMAT(I10,2F10.0)
505      READ(IN,205,ERR=999)((DP(I),SA(I),TA(I),UA(I)),I=1,NPTS)
506      205 FORMAT(4F10.0)
507      ILI=II(5)
508      ILO=ILI
509      IF(ICUTOP .EQ. 0)GO TO 35
510      IDH=1
511      IRKF=1
512      IF(RK)35,35,36
513      35 IRKF=0
514      RK=1.41
515      36 IF(DH)37,37,38
516      37 IDH=0
517      DH=0.1
518      50 CONTINUE
519      38 RETURN
520      40 NW=NPTS+5+ICUTOP

```

PROGRAM ULINE

```

521      IF(LIMIT .EQ. 0 .AND. IDFP .EQ. 1)WRITE(IO,403)
522 403 FORMAT(1H1)
523      IF(IDFP .EQ. 0)GO TO 50
524      WRITE(IO,400)(IFILE(IK),IK=1,19)
525 400 FORMAT(/1X,' UNIVERSAL DATA FILE : ',19A1/)
526      IF(INTER .EQ. 1)WRITE(IO,10)
527 10 FORMAT(
528 *1X,' *****'/
529 *1X,' *   NOTE, THIS IS THE ORIGINAL FILE.          */
530 *1X,' *   IT DOES NOT REFLECT CHANGES MADE INTERACTIVELY. */
531 *1X,' *   THOSE CHANGES ARE SHOWN IN THE OUTPUT HEADING. */
532 *1X,' *****'/)
533      REWIND IN
534      IF(NWO .EQ. 0)GO TO 41
535      DO 1001 I=1,NWO
536        READ(IN,401)(AA(J),J=1,10)
537 1001 CONTINUE
538      41 DO 1000 I=1,NW
539        READ(IN,401)(AA(J),J=1,10)
540        401 FORMAT(10A8)
541        WRITE(IO,402)(AA(J),J=1,10)
542        402 FORMAT(1X,10A8)
543 1000 CONTINUE
544      GO TO 50
545 999 WRITE(IO,403)
546      WRITE(IO,998)(TITLE(IK),IK=1,10)
547 998 FORMAT(1X,' COMPUTATIONS CEASE FOR:'/2X,'CASE I.D. ',10A8/
548 *1X,' INPUT CONVERSION ERROR, CHECK DATA FILE')
549      IF(INTER .EQ. 1)WRITE(IT,998)(TITLE(IK),IK=1,10)
550 99 CALL EXIT
551 END
552 C
553      SUBROUTINE NLINE(K,IFLAG)
554 C
555 C      PRINTS INPUT DATA FOR ULINE IN NEW FORMAT
556 C
557      COMMON/AVAR/DP(30),SA(30),TA(30),UA(30),DENP(30)
558      COMMON/VAR/INTER,IDFP,ILI,ILO,IRKF,QT,NP,PDEP,U,SPACE,NPTS,T,S,
559 *HANG,RHOE,RK,DH,IDH,FR
560      REAL*8 TITLE(10),TITLI(10)
561      BYTE IFILE(20),OFILE(20)
562      COMMON/TVAR/TITLE,TITLI,IFILE
563      COMMON/INOUT/IN,IO,IT
564      DATA IFILE(20)/0/,OFILE(20)/0/
565 C
566      IF(K-2)1,2,3
567      1 IF(IRKF .EQ. 0)WRITE(IO,111)RK
568 111 FORMAT(/2X,'ROBERTS FACTOR SA/SM',16X,'=',F9.2,4X,'(DEFAULT)')
569      IF(IRKF .EQ. 1)WRITE(IO,112)RK
570 112 FORMAT(/2X,'ROBERTS FACTOR SA/SM',16X,'=',F9.2)
571      IF(IDH .EQ. 0)WRITE(IO,113)DH
572 113 FORMAT(2X,'INTEGRATION STEP SIZE',15X,'=',F9.2,4X,'(DEFAULT)',/)
573      IF(IDH .EQ. 1)WRITE(IO,114)DH
574 114 FORMAT(2X,'INTEGRATION STEP SIZE',15X,'=',F9.2,/)
575      IF(T .NE. 0.)WRITE(IO,123)T
576 123 FORMAT(2X,'INITIAL TEMPERATURE OF THE PLUME',4X,'=',F9.2,4X,
577 *'DEGREES CENTIGRADE')
578      IF(T .NE. 0.)WRITE(IO,124)S
579 124 FORMAT(2X,'INITIAL SALINITY OF THE PLUME',7X,'=',F11.4,' PPT')
580      IF(T .EQ. 0.)WRITE(IO,125)S
581 125 FORMAT(2X,'INITIAL DENSITY OF THE PLUME',8X,'=',F12.5,' G/CM3')
582      IF(T .NE. 0.)WRITE(IO,125)RHOE
583      WRITE(IO,126)FR
584 126 FORMAT(2X,'ROBERTS FROUDE NUMBER',15X,'=',F9.2)
585      IF(IFLAG .EQ. 0)GO TO 134

```

PROGRAM ULINE

```

586      WRITE(IO,131)
587      131 FORMAT(/3X,'DEPTH',4X,'SALIN',5X,'TEMP',4X,'DENSITY',6X,'U'/
588      *4X,'(M)',5X,'(PPT)',5X,'(C)',6X,'(G/CM3)',3X,'(M/S)'/)
589      WRITE(IO,132)((DP(I),SA(I),TA(I),DENP(I),UA(I)),I=1,NPTS)
590      132 FORMAT(1X,F7.2,F9.2,F11.5,F9.3)
591      GO TO 2
592      134 WRITE(IO,135)
593      135 FORMAT(/3X,'DEPTH',3X,'DENSITY',5X,'U'/
594      *4X,'(M)',5X,'(G/CM3)',2X,'(M/S)')
595      WRITE(IO,133)((DP(I),DENP(I),UA(I)),I=1,NPTS)
596      133 FORMAT(1X,F7.2,F10.5,F8.3)
597      2 WRITE(IO,200)
598      200 FORMAT(1X)
599      WRITE(IO,201)QT
600      201 FORMAT(2X,'TOTAL EFFLUENT FLOW',17X,'=',F11.4,' CMS')
601      WRITE(IO,202)NP
602      202 FORMAT(2X,'NUMBER OF PORTS',21X,'=',I6)
603      WRITE(IO,203)SPACE
604      203 FORMAT(2X,'PORT SPACING',24X,'=',F10.3,' M')
605      WRITE(IO,204)HANG
606      204 FORMAT(2X,'HORIZONTAL ANGLE',20X,'=',F8.1,5X,'DEGREES')
607      WRITE(IO,205)PDEP
608      205 FORMAT(2X,'PORT DEPTH',26X,'=',F9.2,' M')
609      RETURN
610      3 WRITE(IO,210)(TITLI(IK),IK=1,10)
611      210 FORMAT(/2X,66('X'))//2X,'RUN TITLE: ',10A8/)
612      WRITE(IO,126)FR
613      GO TO 2
614      END
615  C
616      SUBROUTINE USORT(DP,X,Y,Z,NPTS)
617  C
618      DIMENSION DP(30),X(30),Y(30),Z(30)
619      NESTED=NPTS
620      L=NESTED-1
621      DO 1000 M=1,L
622          NESTED=NESTED-1
623          DO 1000 I=1,NESTED
624              IF(DP(I) .LE. DP(I+1))GO TO 1000
625              DUMMY=DP(I)
626              DP(I)=DP(I+1)
627              DP(I+1)=DUMMY
628              DUMMY=X(I)
629              X(I)=X(I+1)
630              X(I+1)=DUMMY
631              DUMMY=Y(I)
632              Y(I)=Y(I+1)
633              Y(I+1)=DUMMY
634              DUMMY=Z(I)
635              Z(I)=Z(I+1)
636              Z(I+1)=DUMMY
637      1000 CONTINUE
638      RETURN
639      END
640  C
641      FUNCTION SIGMAT(SAL,T)
642  C
643      SIGO=((6.8E-6*SAL)-4.82E-4)*SAL+.8149)*SAL-0.093
644      B=1.E-6*T*((.01667*T-.8164)*T+18.03)
645      A=.001*T*((.0010843*T-.09818)*T+4.7867)
646      SUMT=(T-3.98)*(T-3.98)*(T+283.)/(503.57*(T+67.26))
647      SIGMAT=(SIGO+.1324)*(1.-A+B*(SIGO-.1324))-SUMT
648      RETURN
649      END
650  C

```

PROGRAM ULINE

```

651      FUNCTION CALC(F,K)
652  C
653      COMMON/XX/FF(4,3),AA(4,3),BB(4,3),ZZ(3),NN(3)
654      DATA FF/.2,13.84,30.,100.,.13,.3,.6,100.,.07,.13,100.,0./
655      DATA AA/.27,.3167,.2387,.17,.27,.3536,.3936,.4125,.27,
656      *.4367,.5879,0./
657      DATA BB/.3333,.2346,.127,.02723,.3333,.2020,.1131,.0213,.3333,
658      =.1525,.0067,0./
659      DATA ZZ/.150,.374,.570/
660      DATA NN/4,4,3/
661      N=NN(K)
662      DO 1000 I=1,N
663          IF(F-FF(I,K))1,1,1000
664      1000 CONTINUE
665          CALC=ZZ(K)
666          RETURN
667      1 CALC=AA(I,K)*(F**(-BB(I,K)))
668          RETURN
669      END

```